ACM SIGGRAPH 2001 Course #26

Internetworked 3D Computer Graphics: Overcoming Bottlenecks, Supporting Collaboration and Stepping up to Wireless Connectivity

Course Organizers:

Theresa Marie Rhyne North Carolina State University

Bob Barton Fraunhofer Center for Research in Computer Graphics

Instructors:

Don Brutzman Naval Postgraduate School

Grantley Day Maxis Web Group of Electronic Arts

Mike Macedonia
USA Simulation, Training, and Instrumentation Command (STRICOM)

Abstract: Networking and computer graphics are distinct disciplines that are converging to support 3D collaborative exploration, visualization, multi-user video games and large-scale virtual environments (LSVEs). This course presents an introduction to networking concepts for using and developing interactive and collaborative Internet-based graphics applications. It also addresses criteria for streaming 3D graphics in mobile and wireless computing situations. Key challenges such as interactivity, latency, content distribution, bandwidth, packet management, synchronization, geometry streaming, large world database management and personalization will be highlighted.

ACM SIGGRAPH 2001 Course #26 Notes

Internetworked 3D Computer Graphics: Overcoming Bottlenecks, Supporting Collaboration and Stepping up to Wireless Connectivity

Introduction:

The concept of Internetworked Graphics is used here to describe the merger and dependencies of computer graphics applications and the networking infrastructure. During this course, we will provide an overview of network infrastructure issues relating to interactive 3D graphics. Relevant networking capabilities, available to all users, will, hopefully, be demonstrated in real time. We show how the Mulitcast Backbone (MBone) functions to support real time video and audio. Collaborative work tools that incorporate computer graphics functionality across the Internet are also be discussed and demonstrated. A featured case study highlights the challenges faced with implementing and maintaining an online computer game. The relationship of OpenGL, VRML, X3D, Java/Java3D, MPEG, HTML, XML, and SMIL to Internet, streaming and Distributed Interactive Simulation (DIS) protocols is reviewed. We also cover High Level Architecture (HLA) and Wireless Application Protocol (WAP) implementations.

We will highlight how the development of the next generation of Web software is dependent on effective implementation of Internetworked Graphics concepts. Those developing networking protocols and MBone capabilities will need to factor in the impact that the next - generation Virtual Reality Modeling Language (VRML), interactive visualization, and large scale virtual environments (LSVE) methods have on future bandwidth and latency requirements. Those creating interactive graphics tools for the Web and the Internet will want to understand the underlying networking standards and infrastructure in order to build effective 3D collaborative applications. Integrated approaches allow us to move beyond the bottlenecks and roadblocks of Internetworked Graphics.

We hope you enjoy participating in this course and reading these notes.

Theresa Marie Rhyne, Bob Barton, Don Brutzman, Grantley Day and Mike Macedonia

Internetworked 3D Computer Graphics Course Outline

I. Introductory Remarks -- Rhyne

Motivation and expected outcomes. Organization of course and demos. (5 minutes)

II. Overview of 3D Interactive Graphics using the Internet -- Rhyne (30 minutes)

A big-picture description of the key issues and practical aspects of internetworked graphics and collaborative visualization across the Internet are reviewed. Six components of Internetworked Graphics are introduced: connectivity, content, interaction, economics, applications and personalization. We highlight the interaction of 3D graphics with streaming video and audio technologies. This includes RealNetworks, Windows Media, SMIL, QuickTime, and Shockwave applications. Several examples of information visualization and virtual reality are presented that illustrate how 3D graphics can be used to help solve key problems in global networking. We show that interactive graphics and visual information retrieval can provide new insight into key web and network problems such as topology visualization, web searching, performance optimization and network monitoring. Internetworked graphics is a two-way collaboration that can significantly benefit by cooperative work between the computer graphics and networking communities.

III. The Realities of Implementing Internetworked 3D Graphics -- Day (45 minutes)

A case study presentation of the challenges faced with designing, building, implementing and maintaining a commercial Internetworked 3D Graphics application is presented. The Sims "game" from Electronic Arts - Maxis was launched in 1999, (http://www.thesims.com). The realities of handling the associated online Web side, server hardware, operating system, security, hacking, cheating, changing infrastructure, and continued customer support will be covered in this discussion. Future trends associated with new games under development and their online presence will also be addressed.

Morning Break

IV. Internetworked Graphics: Capabilities, Shortfalls, Frontiers -- Brutzman (45 minutes)

Networking and graphics are two halves of an interlocking whole: large-scale virtual environments (LSVEs). This lecture will reemphasize the six components of Internetworked 3D Graphics (connectivity, content, interaction, economics, applications and personal impacts) with open source code implementations. Concepts to be presented include the Internet Protocol (IP), unicast/broadcast/multicast, http, streaming protocols, MBone audio/video/whiteboard/image streams, the Distributed Interactive Simulation (DIS) protocol, behavior protocols, etc. Recent work to support the integration of DIS-Java-VRML along with real time demonstrations of applications that use DIS-Java-VRML pubic domain software will be highlighted. The notion of a Virtual Reality Transfer Protocol (vrtp) will be introduced and described. The proposed concept of vrtp is designed to support VRML in the same manner that http supports HTML. (See: http://www.stl.nps.navy.mil/~brutzman/vrtp/). We will show how to tie network bottlenecks to

MBONE and Virtual Environment Demonstration - Brutzman and all instructors. (30 minutes)

performance just as we evaluate graphics bottlenecks.

In order to maximize value to course attendees, we will demonsrate a series of example programs of immediate practical use for networking graphics. We will use the DIS-Java-VRML library

(http://www.web3d.org/WorkingGroups/vrtp/dis-java-vrml/) to show how ordinary VRML scenes can be easily composed and internetworked. We will compare & contrast Java+VRML with a Java3D example. We will also collaborate with remote sites via the Multicast Backbone (MBone), showing public domain audio, video and behavior streams together with 3D VRML. Source code and web pointers will be provided throughout.

Lunch

V. Integrating IETF Standards with 3D Programming Paradigms & Issues in Distributed Virtual Environments-- Barton (45 minutes)

The impact of developing standards such as IETF, ITU, MPEG on developers and content creators using OpenGL, Java3D, VRML, X3D, MPEG-4, MPEG-7 and MP3 will be highlighted. This year's focus presentation will include an overview of MPEG, and the latest developments within MPEG-4 and MPEG-7. Issues and applications relating to distributed visualization and virtual environments are discussed. Attendees will learn how network infrastructure and programming issues integrate with real time internetworked 3D graphics applications.

Collaborative Virtual Reality Demonstration - Barton (30 minutes)

What can virtual environment developers learn from object oriented and component based programming? Baobab is a component framework for the development of interoperable content and simulations for virtual worlds. Baobab combines an extensible simulation API with a distributed knowledge database to allow developers to compose complex simulations that are aware of and can interact with their surrounding environment. We will demonstrate the ease with which these components can be developed, their expressive power and the emergent behavior resulting from their dynamic nature.

Afternoon Break

VI. Distributed Experiences -- Macedonia (45 minutes)

Broadband communications, home networks, and powerful 3D game console technology are reshaping entertainment and potentially the delivery of education. Building upon the IETF discussion and the collaborative virtual environment demonstration, we provide an overview of issues and a taxonomy of distributed environments that provide entertainment, collaboration and experiential learning. The tradeoffs between bandwidth, latency, frame rate, packet loss, and router congestion, will be reviewed. Current efforts of researchers to develop distributed environments and the importance of interaction with the entertainment community will be discussed. This discussion includes research into immersive techniques to provide emotional experiences in distributed environments for applications such as Advanced Distributed Learning (ADL).

Distributed Virtual Environment and Game Demonstrations - Macedonia and all instructors. (30 minutes)

Practical demonstrations of networked simulators used for skill-based and cognitive level learning will be shown. Applications will be demonstrated and reviewed in real time. We will include computer games examples and build up to training on distributed networked simulators. The demonstrations includes the use of commercial PC games used to stimulate interaction with real-world systems for training as well as new game console technology such as the Microsoft X-Box.

Instructors' Biographical Information:

Bob Barton
Fraunhofer Center for Research in Computer Graphics
321 South Main Street
Providence, Rhode Island 02906 USA
401 453-6363 x111 voice, 401 453-0444 fax
rjbarton@crcg.edu

Robert J. Barton III is the Project Director of Transatlantic Research and Development Environment (TRADE) Program, and Department Leader for Global Work Environments at the Fraunhofer Center for Research in Computer Graphics (in Providence, Rhode Island. His primary responsiblties include directing the development of distributed collaborative environments in support of TRADE, a designated G7 testbed project for enabling technologies in global electronic business. Bob's engineering career has included engineering billets on US Naval Surface ships, instructor duty at the Naval Academy, and systems engineering branch head for the Trident Missile Program of Strategic Systems Programs Office, Sunnyvale, CA. His areas of research include VLSI High-radix arithmetic circuits, computing the average number of nodes of Ordered Binary Decision Diagrams (OBDDs), and the modeling and analysis of multicast network protocols.

Don Brutzman Code UW/Br, Naval Postgraduate School Monterey California 93943-5000 USA 831.656.2149 voice, 831.656.3679 fax brutzman@nps.navy.mil http://www.stl.nps.navy.mil/~brutzman

Don Brutzman is an Assistant Professor in the Interdisciplinary Academic Group at the Naval Postgraduate School. His research interests include underwater robotics, real-time 3D computer graphics, artificial intelligence and high-performance networking. He has highlighted networking issues in panel presentations at SIGGRAPH 94 & 95 and organized the "VRML: Prelude and Future" panel at SIGGRAPH 96. He currently serves as the Web3D Consortium's Vice President for Technology and was the Conference Chair for VRML 97 & 98 as well as Web3d-VRML 2000. Currently, he is the SIGGRAPH 2001 Online Chair. Brutzman also authored the chapter on "Internetworked Graphics: Bottlenecks and Breakthroughs" that appears in Digital Illusions, (Clark Dodsworth - editor), Addison Wesley, 1997. Don is a member of the Institute of Electrical and Electronic Engineers (IEEE), ACM SIGCOMM & SIGGRAPH, and the American Association for Artificial Intelligence (AAAI).

Grantley Day Maxis: A Division of Electronic Arts Suite 600 2121 N. California Blvd Walnut Creek, California 94596-3572 gday@maxis.com

Grantley Day has been involved in the computer game industry for the past seven years, focusing specifically on the Internet and how it relates to computer entertainment for the past four. Before working at Electronic Arts, Grantley was the founder of Chaos Concepts, an Australian based software

company that developed the successful multimedia title Entombed. Since re-locating to the United States in late 1995, Grantley worked for two startup companies before joining Electronic Arts in 1997. At a time when the Internet was becoming increasingly important, Grantley was the lead architect on a number of technologies to allow Electronic Arts' entertainment titles to communicate across the Internet.

Grantley is currently a Director of Development at Maxis, a division of Electronic Arts, where he heads the web group, focusing on the Internet features of key titles such as The Sims and SimCity. His most recent effort was architecting, designing and leading the successful launch of TheSims.com, the Internet component of this year's hit game The Sims. TheSims.com is highly integrated with the game and provides the venue for players to share their experience and creations. TheSims.com has been an instant success with close to 80,000 unique visitors accessing the site each day to trade Sims families, read stories created and posted by other users and participate in community events.

Michael R. Macedonia USA Simulation, Training, and Instrumentation Command (STRICOM) 12350 Research Parkway Orlando, Florida 32826-3276 voice: (407) 384-3500 macedonia@computer.org

Michael R. Macedonia is Chief Scientist and Technical Director for the USA Simulation, Training, and Instrumentation Command (STRICOM). He has worked extensively with the multicast networking community and has written one of the most often referenced publications in that area. He also contributed to the National Academy of Sciences report entitled "Virtual Reality: Scientific and Technological Challenges" detailing the networking and communications research needed to continue the development of virtual reality systems. At SIGGRAPH 96, he was a panelist on the "Global Multi-User Virtual Environments" panel and at SIGGRAPH 2000, was a panelist on "Emotional Simulator: The Tears and Fears of Creating a Compelling Simulated Experience for Both Entertainment and Training.

Theresa Marie Rhyne 3418 Balfour East Durham, North Carolina 27713 919 544-1817 voice, rhyne@siggraph.org

Theresa Marie Rhyne is currently a multimedia/visualization expert in Learning Technology Services at North Carolina State University. In the early part of 2001, she was an independent consultant in 3d computer graphics and visualization. From 1990 –2000, she was a government contractor (initially for Unisys Corporation (1990 – 1992) and then for Lockheed Martin Technical Services (1993 – 2000)) at the US EPA Scientific Visualization Center. She was the founding visualization expert at the Center. From 1996 - 2000, she was a Director-at- Large on ACM SIGGRAPH's Executive Committee and initiated the SIGCOMM - SIGGRAPH collaboration that resulted in cross-ACM SIG tutorials on Internetworked 3D Computer Graphics at SIGCOMM 97, SIGGRAPH 97, SIGGRAPH 98, SIGGRAPH 99 and SIGGRAPH 2000. She continues to be project director of the ACM SIGGRAPH - SIGCOMM Collaboration, the ACM SIGGRAPH Outreach to the Computer Games Community and the ACM SIGGRAPH Carto Project. Her research interests include visualization toolkit development, collaborative computing in a networked environment, and the integration of geographic information systems with scientific visualization methods. Recently, Theresa-Marie has explored the influence computer games are having on scientific and information visualization.

ACM SIGGRAPH 2001 Course #26 Notes

Internetworked 3D Computer Graphics: Overcoming Bottlenecks, Supporting Collaboration and Stepping up to Wireless Connectivity

II. Overview of 3D Interactive Graphics using the Internet

Theresa Marie Rhyne
Independent Consultant
3D Computer Graphics Technologies
Email: rhyne@siggraph.org

Introduction:

II. - 1: Defining Internetworked 3D Graphics

Although networking and computer graphics are considered to be distinct disciplines, these fields begin to converge in order to support collaborative exploration and visualization on the Internet and the World Wide Web (Web). Telecommunication breakthroughs remove bottlenecks and provide new opportunities for interactive three-dimensional (3D) graphics across globally interconnected dissimilar networks. Multicast Backbone (MBone) tools, developed in the networking arena, provide desktop videoconferencing tools and the potential for streaming graphics to facilitate the sharing of visualization and virtual reality explorations. 3D Web technologies, like the Virtual Reality Modeling Language (VRML) and Rich Media 3D (RM3D), support the interactive display and fly through of networked computing resources on the Internet. The emergence of multi-player computer games, chat rooms, real time Webcasting, and streaming audio and video across the Internet acutalizes this ongoing convergence. The delivery of rich streaming media content on mobile phones opens up possibilities for streaming interactive 3D computer graphics across wireless protocols. Here, we define the concept of "Internetworked 3D Graphics" to describe the continued merger and dependencies of these two fields.

Reference: T-M. Rhyne, D. Brutzman and M. Macedonia, "Internetworked Graphics and the Web", (under Internet Watch: Ron Vetter, editor), Computer, IEEE Computer Society, Vol. 30, No. 8 (August 1997), pp. 99 - 101.

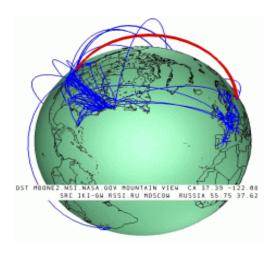


Figure # II-1: Example of the use of the Virtual Reality Modeling Language (VRML) to visualize network data. This snapshot shows the topology of the MBone mapped on to a 3D globe. Image courtesy of Tamara Munzner at Compaq SRC. Reference: Tamara Munzner, Eric Hoffmann, K. Claffy, and Bill Fenner, "Visualizing the Global Topology of the MBone", Proceedings of the 1996 IEEE Symposium on Information Visualization, October 28-29, 1996, San Francisco, California, pp. 85 - 92. (http://www-graphics.stanford.edu/papers/mbone)

II - 2: Six Components of Internetworked 3D Graphics

Six aspects of Internetworked 3D Graphics include: **connectivity, content, interaction, economics, applications** and **personal impacts**. **Connectivity** refers to the capacity, bandwidth, protocols, and multicasting capabilities on the networking infrastructure. Just as OpenGL, VRML/X3D, and RM3D are 3D computer graphics specifications, a networking protocol is a set of rules or procedures for governing data exchange on the Internet. The Internet Engineering Task Force (IETF) constitutes the technical community associated with many networking protocols. The IPv6 Suite is the current set of networking protocols. New telecommunications infrastructure is beginning to emerge to support general user access to Internetworked 3D Graphics. Home networking services, such as digital subscriber lines (DSL), are starting to deliver rapid (T1) speeds. Other end-user networking services support wireless networking. More information on home networking is available at the Web ProForum site of the International Engineering Consortium, see: (http://www.webproforum.com/home_net/#).

Reference: Robert E. Fliman, "Embedded Internet Systems Come Home" (Editor's Introduction), IEEE Internet Computing, Institute of Electrical and Electronic Engineers (IEEE), Vol. 5, No. 1, Jan/Feb 2001, pp. 52 – 53.

Content encompasses any type of information, dataset or stream that is transported via a networking protocol. The VRML example shown in Figure #II-1 is an example of content. Other web technologies can also provide Internetworked 3D content. For example, the Motion Pictures Experts Group (a working group of the International Organization for Stardards - ISO) developed and is updating the standard for multimedia content via the Web (entitled MPEG-4). MPEG-4 technology allows for streaming 3D graphics content and mixed media scenes on the Web. The Synchronized Multimedia Integration Language (SMIL) is another tool that facilitates the writing of multimedia presentations on the Web. Flash, from Macromedia (http://www.macromedia.com), and LiveMotion, from Adobe (www.adobe.com), are two of the more popular ways to transmit moving images that play at

respectable rates across the Internet. Current trends in online video content include video clips that can be rotated 360 degrees as they are playing.

Interaction involves minimal latency, a sense of presence, and the ability to both access and modify content. In general, the key bottleneck to collaborative visualization and large scale virtual environments (LSVEs) is not 3D graphics rendering. It is telecommunications networking. The IEEE Distributed Interactive Simulation (DIS) Protocol is used to communicate position, orientation, velocities and accelerations of multiple entities in a networked environment. Combining DIS, Java and VRML technologies together allows for determining interaction between networked 3D objects. The High Level Architecture (HLA) for distributed virtual environments is another specification effort underway to address interaction issues.

The **Economics** of Internetworked 3D Graphics includes the expense of access and use of connectivity, content and interaction as well as methods of handling Web transactions. Historically, the cost of interactive LSVEs and collaborative visualization was coupled with funding for high speed networking. As the Web and 3D content expand, the demand to provide affordable multi-user interaction for general purpose Internet users increases. The continued rise of electronic commerce (e-commerce) via the Web will increase the demand to develop streaming and interactive 3D content for Web advertising. An example of economic factors can be found by viewing the AOL.COM pricing plans at (http://www.aol.com/info/pricing.html). Another example includes Sony's e-distribution model for its PlayStation2 gaming consoles. An add-on modem or broadband/Ethernet connector is expected to become available, in late 2001, that facilitates chat, email, shopping and online gaming at a PlayStation2 console.

Applications establish the context for Internetworked Graphics and determine requirements for infrastructure development. Examples of applications include 3D games, interactive instruction, scientific experiments, and distributed simulations in a telecommunications environment. **Personal Impacts** involve the sense of wonder (and maybe frustration) associated with Internetworked information discovery and exploration. Intelligent agent and other technologies are evolving to assist with personalization needs on the Internet.

Reference: Don Brutzman, "Graphics Internetworking: Bottlenecks and Breakthroughs," chapter in Digital Illusions, Clark Dodsworth (editor), Addison-Wesley, ACM SIGGRAPH Press, 1997. Available at: http://web.nps.navy.mil/~brutzman/vrml/breakthroughs.html).

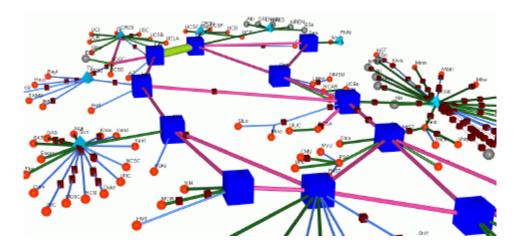


Figure # II-2: Example output from an Open-GL based "visualization engine" for examining IP address utilization, network usage based on a network address matrix and bucketed packet length distributions.

This application, entitled "Cichlid", was developed and is maintained by the National Laboratory for Applied Network Research (NLANR)'s Measurement & Operations Analysis Team (MOAT) with support from the National Science Foundation (NSF), located at the San Diego Supercomputer Center (SDSC). Cichlid is freely distributed software. More information on Cichlid can be found at: http://moat.nlanr.net/Software/Cichlid/). The Third quarterly report (July 2000 - September 2000) of NLANR's MOAT efforts provides a good discussion of the rationale behind and uses of Cichlid as a network visualization tool, (http://moat.nlanr.net/Reports/MOAT-3Q2000/). Image shown courtesy of Jeff Brown and Hans-Werner Braun of NLANR's Measurement & Operation Analysis Team (MOAT).

II - 3: MBone - Internetworked Video & Audio Communication

To build on the six aspects of Internetworked 3D Graphics decribed above, we will first describe a highly successful effort to provide open desktop videconferencing on the Internet. This application was developed primarily by researchers from the telecommunications community.

The multicast backbone (MBone) is a connectivity application on the Internet. It is used for live audio, video and other network packets on a global scale. MBone is a virtual network because it shares the same physical media as the Internet. It uses a network of routers (mrouters) that can support multicast. These mrouters are either upgraded commercial routers, or dedicated workstations with modified operating system kernels running in parallel with standard routers. Two things make multicasting feasible on a world-wide scale: installation of high bandwidth Internet backbone connections, and widespread availability of computers with adequate processing power and built-in audio capability.

Multicast provides one-to-many, several-to-many, and many-to-many network delivery services. A one-to many example might be a live transmission of a technical conference. A several-to-many might be a panel interview between different locations attended by many disperse people. A many-to-many example might be a distributed research exercise with 100 active contributors. Multicast is useful for a variety of videoconferencing, audio and multiplayer events where numerous hosts need to communicate simultaneously.

MBone software tools are in the public domain. An original suite of tools was developed for UNIX workstations by researchers at Lawrence Berkeley National Laboratory. The Microsoft Bay Area Research Center has subsequently developed freeware MBone tools for Windows 95, 98 & NT platforms. Multicasting videoconferencing applications for Linux have also evolved. Merit Network Inc. provides a listing of MBone software for various computing platforms. We note these respective web sites below.

Lawrence Berkeley National Laboratory - MBone Info. (http://www-itg.lbl.gov/mbone/).

Microsoft Research - Mbone Info for Windows 95, 98 & NT (http://www.research.microsoft.com/research/BARC/mbone/).

Linux Multicast Information by Mike Esler (http://www.cs.washington.edu/homes/esler/multicast/#apps)

Merit Network Inc. - MBone Software Archives (http://www.merit.edu/~mbone/index/titles.html).

Reference: Steven McCanne, "Scalable Multimedia Communication Using IP Multicast and Lightweight

Sessions", IEEE Internet Computing, Institute of Electrical and Electronic Engineers Inc. (IEEE), Vol. 3, No. 2, March/April 1999, pp. 33 - 45. See electronic version of article abstract at: (http://www.computer.org/internet/ic1999/w2033abs.htm).

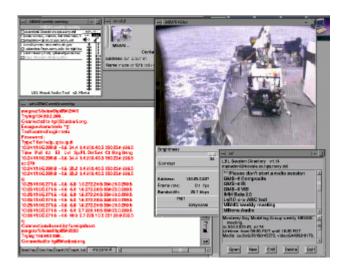


Figure # II-3: Example of an MBone session showing application tools nv (network video), vat (visual audio tool), wb (whiteboard), and sd (session directory). This session occurred at the Monterey Bay Aquarium Research Institute. Image courtesy of Don Brutzman at the Naval Postgraduate School.

Researchers are Lawrence Berkeley National Laboratory have expanded their research and are now developing software tools to facilitate remote control capabilities for Internet based videoconferencing. These tools permit users to "walk" around the room, focus in on objects, and actively participate in a videoconference rather than just observe. More information on these efforts can be found at the following Web site: (http://www-itg.lbl.gov/mbone/remote/). Another forward looking videoconferencing notion includes the National Tele-Immersion Initiative (NTII) where efforts are underway to use multiple cameras, laser rangefinders, large screens and head-tracking gear to deliver a high-resolution 3D view of the videoconferencing (workspace) environments. Since there are no existing 3D-transmission standards, NTII researchers designed a customized protocol to replace the Transmission Control Protocol (TCP). For more information on NTII, see their web site at: (http://www.advanced.org/tele-immersion/). TCP is highlighted in the next section of these course notes.

II - 4: Examining Internet Protocol (IP) Unicasting, Multicasting & Streaming

At a fundamental level, the Internet is based on a protocol, the Internet Protocol (IP), that does not guarantee the delivery of any particular packet of data. Another protocol, defined as the Transmission Control Protocol (TCP), compensates for this potential loss of packets. TCP adds careful error correction by marking data with sequence numbers. Any data that appears is retransmitted by TCP. Together, TCP/IP form a reliable connectivity solution that guarantees data will be delivered in its proper order, no matter how long it takes.

Most traditional Internet Protocol (IP) applications, such as Web browsing and E-Mail, employ unicasting to transmit data. Unicasting is where a separate connection is set up between a server and each of its clients. This type of connectivity is optimal for situations where every client has specific needs such as browsing Web pages and reading particular E-Mail messages. For collaborative situations, where multiple clients all want the same data at the same time, unicasting is inefficient since identical streams of data are sent to each client. This is a waste of both server and network capacity. So, another

connectivity solution, multicasting, is more optimal.

Multicasting allows a server to transmit only a single data stream, regardless of how many clients might request it. Whenever this data stream traverses a multicast-enabled switch or router, it is copied. Copies are only made to branches of the networking tree where clients that requested the data stream are located. This results in a dramatic reduction in overall bandwidth consumption. Multicasting is ideal for situations where clients want the identical data simultaneously. Live video and audio streams are examples of these situations. The MBone (discussed in the previous section of these notes) is an application that provides this type of desktop videoconferencing.

Multicasting is being promoted by a telecommunications industry consortium of 100 companies that was founded in 1996. This consortium is entitled the IP Multicast Initiative (IPMI). Cisco Systems, 3Com, Lucent Technologies, GTE Internetworking, and Stardust Technologies are among the consortium's members. More information on IPMI can be found at: (http://www.ipmulticast.com/).

Since multicasting uses only IP, data packets can become lost. The MBone is sometimes called a loss-tolerant application since it permits temporary losses in audio or video tranmissions across the Internet. In some multicasting situations, a feedback or checking mechanism is desirable. The Real-Time Transfer Protocol (RTP) and the Real-Time Transport Control Protocol (RTCP) together provide a solution to this concern. RTP and RTCP run on top of the User Datagram Protocol (UDP) to provide timing information necessary to synchronize and display audio and video data.

The Real Time Streaming Protocol (RTSP) is another protocol relevant to multicasting. RTSP was first published as a proposed standard in April 1998 by the Internet Engineering Task Force (IETF). RTSP acts as a framework for controlling multiple data delivery sessions and assists with switching between TCP, UDP and RTP sessions as needed. RTSP helps with firewall configurations and provides VCR-style controls such as pause, fast-forward, reverse or seek for previously recorded data streams. The fundamental difference between TCP/IP and RTSP can be simply stated. A real-time streaming protocol (RTSP) is most concerned about when a data packet arrives while TCP/IP is concerned about if the packet has arrived. Researchers at RealNetworks Inc. developed the basic framework for RTSP.

Efforts are also underway to establish protocols for access to the Internet from digital mobile phones, pagers, personal digital assistants, and other wireless terminals. In mid-1997, mobile computing vendors began work on the Wireless Application Protocol (WAP). WAP is becoming a defacto standard. More information on WAP is available at: (http://www.wapforum.org/who/index.htm).

Multicasting, RTSP, and WAP are important connectivity building blocks for Internetworked 3D Graphics. Live Webcasting is optimized with these networking protocols. With these standards in place, we can now begin to consider file formats for supporting streaming media across the Internet. Example formats include RealMedia (from RealNetworks Inc.), QuickTime (from Apple Computer Corporation), Windows Media Advanced Streaming Format (from Microsoft Corporation), and Viewpoint Corporation's 3D file format (formerly know as Metastream.com). We note the respective web sites for these tools below. The upcoming sections of these course notes highlight the SMIL, MPEG-4, and RM3D standards, which are intended to support streaming audio, video and 3D content across the Web.

RealNetworks Inc. - Streaming Media Web Site (http://www.realnetworks.com/)

Apple Computer, Inc. - QuickTime Web Site (http://www.apple.com/quicktime/)

Microsoft Corporation - Windows Media Technologies

(http://www.microsoft.com/windows/windowsmedia/en/features/roadmap.asp)

Viewpoint Corporation's (formerly Metastream) Web Site

(http://www.viewpoint.com/create/e4e.html)

II - 5: Streaming Media and SMIL

The concept of streaming media is based on the notion that it is not necessary to completely download content from the Internet before it is played. Streaming audio is likely one of the most widely known applications of streaming media across the Web. In 1995, RealNetworks Inc. released their RealPlayer plug-in to facilitate real-time or streaming audio via Web browsers. The company expanded to address real video with their streaming media plug-ins. As of 2000, there were three leading streaming media players: RealPlayer, from RealNetworks, had over 130 million users. Windows Media, from Microsoft, had over 100 million users. QuickTime, from Apple Computer, has 50 million users. For strictly streaming audio purposes, MP3 had 50 million users. Nokia, the mobile handset supplier, will offer RealNetworks' RealPlayer tool on its EPOC-based communicators and smart phones. This and other similar announcements provide opportunities for rich streaming media content delivery to mobile phone and wireless computing devices. The mobile Internet computing market has the potential to grow to 1 billion users by 2005.

In addition to providing players for streaming audio and video, RealNetworks also provides RealSystem G2 for encoding, combining, serving and delivering streaming content across the Web. QuickTime Streaming Server 3 is another alternative for serving streaming video across the Internet. These course notes will not go into the details of encoding and hosting streaming media. An excellent resource on these topics is the forth coming book from Addison-Wesley "e-Video: How to Produce Internet Video as Broadband Technologies Converge" by H.P. Alesso. Alesso's Video Software Laboratory Web site is also an effective resource.

Video Software Laboratory - Encoding & Hosting Streaming Video

(http://www.video-software.com/)

Reference: H.Peter Alesso, e-Video: How to Produce Internet Video as Broadband Technologies Converge, Addison-Wesley, 2000.

In order to understand how to assemble an integrated multimedia streaming presentation, we will highlight the Synchronized Multimedia Integration Language (SMIL). SMIL is a markup language for describing the temporal behavior, screen layout and associated hyperlinks of a streaming media presentation. SMIL is a specification based on the Extensible Markup Language (XML). XML is a technology for supporting richly structured documents over the World Wide Web (Web). The XML specification was fostered by the World Wide Web Consortium and is emerging as a technology for next-generation Web development, (see: (http://www.w3.org/TR/REC-xml)).

In the code example we show below, SMIL is used to combine a GIF image (.gif), hypermedia links (links.rt), streaming text (ticker.rt), streaming audio (voice1a.rm), and streaming video (video.rm) into one integrated presentation. Figure #4 shows a screen capture of this multimedia composition. Multimedia servers need to be configured to deliver SMIL presentations. RealServer, from RealNetworks, supports SMIL files.

The SMIL Specification for the World Wide Web Consortium

(http://www.w3.org/TR/REC-smil/)

```
<smil>
<head>
<meta name="title" content="Rhyne"/>
<meta name="author" content="Video Software Laboratory"/>
<meta name="copyright" content="(c) 2000"/>
<layout>
<root-layout width="470" height="368" background-color="black"/>
<!-- Logo Region -->
<region id="logo" left="4" top="2" width="200" height="100"</pre>
fit="fill" z-index="2" />
<!-- Links Region -->
<region id="headlines" left="16" top="102" width="186" height="30"</pre>
fit="fill" background-color="black" z-index="2" />
<region id="links" left="16" top="102" width="186" height="84"</pre>
fit="fill" background-color="#160f60" z-index="3" />
<!-- Video region -->
<region id="video_background" left="210" top="3" width="240" height="262"</pre>
background-color="black" z-index="1"/>
<region id="video_border" left="188" top="3" width="184" height="153"
fit="fill" background-color="black" z-index="2" />
<region id="video" left="208" top="3" width="256" height="192"</pre>
background-color="black" z-index="3" />
<!-- Ticker Region -->
<region id="last_updated" left="16" top="200" width="393" height="20"</pre>
fit="fill" background-color="#160f60" z-index="2" />
<region id="ticker_background" left="16" top="208" width="393" height="25"</pre>
background-color="black" z-index="2"/>
<region id="ticker" left="16" top="216" width="393" height="115"</pre>
z-index="2" />
</layout>
</head>
<body>
<par>
<!-- Play these streams concurrently (in parallel). -->
<a href="http://www.mpeg2-video.com/" coords="0,0,15,15" show="new" >
<img src="pix/svc_labs2.gif" region="logo" fill="freeze"/></a>
<text src="text/title_last_updated.rt" region="last_updated" fill="freeze" />
<text src="text/links.rt" region="links" fill="freeze" />
<video src="video/video.rm" region="video" fill="freeze" />
<audio src="audio/voicela.rm" />
<text src="text/ticker.rt" region="ticker" />
```



Figure # II-4: Snapshot of the streaming content that results from the Synchronized Multimedia Integration Language (SMIL) code shown above. Peter Alesso of Video Software Laboratory developed this example for presentation in this course at SIGGRAPH 2000 & 2001, (http://www.video-software.com/). For more information on the geographic visualization content highlighted, see: (http://www.epa.gov/gisvis/texture).

II - 6: Metastream - Streaming 3D objects over the Internet

Stepping beyond streaming video, efforts are also underway to stream 3D objects over the Internet. Viewpoint Media Player is a streaming tool for this purpose. Originally named Metastream, it was developed in a joint project with MetaCreations and Intel. Viewpoint (or Metastream) objects stream across the Internet without a server and scale automatically to support the end-user's processing power. With the aid of smart compression technology, users with networking connection speeds ranging from a 56K modem to a T1 (or higher) line are able to examine 3D content in real time. Viewpoint provides a free Web browser plug-in to view content. A modeling tool is required to create the 3D objects.

The Viewpoint Web Site (http://www.viewpoint.com)





Figure # II-5: Snapshot from a Viewpoint or Metastreaming example of "Shared Architecture". Researchers at the Centre for Advanced Spatial Analysis of the University College London are using Viewpoint (or Metastream) technology in their Online Planning and Internet based visualization research. More information on these projects is available at: (http://www.casa.ucl.ac.uk/public/meta.htm). To learn how the architecture models evolved from photographs, see: (http://www.casa.ucl.ac.uk/public/guidelines.htm). Image shown courtesy of Andy Smith.

II - 7: MPEG-4 - Multimedia Standard

The Moving Pictures Expert Group (MPEG) of the International Organization for Standardization (ISO) is charged with developing standards for code representation, processing, compression and decompression of moving pictures, sound and their combination (multimedia). MPEG-1, introduced in 1992, plays out video and audio in linear streams and operates like a digital video player. MPEG-2, introduced in 1995, supports compression and transmission of digital television signals. MPEG-4 is a multimedia standard that allows users to interact with objects within a scene. MPEG-4 (version 1) was introduced in late 1998 while MPEG-4 (version 2) was approved in December 1999. MPEG-7 is a content representation standard for information searching that is still under development with an expected release in 2001. MPEG-21 is a new effort to define a Multimedia Framework to support the delivery of electronic content that began in December 1999. The MPEG Web site is located at: (http://www.cselt.it/mpeg/).

The MPEG-4 standard defines audio and visual components as objects. When multiple objects are grouped together a composition is defined that results in an MPEG-4 scene. Visual objects for a scene are described mathematically and given a position in 2D or 3D space. Audio objects are placed in a sound space. For 3D space, content only needs to be defined once. When a user changes their viewpoint, the calculations to update the screen and sound are performed locally at the user's computer. This allows for interaction with a MPEG-4 scene.

The Binary Format for Scenes (BIFS) is MPEG-4's language for defining and changing scenes. BIFS commands add and delete objects as well as changing visual and acoustic properties of objects. BIFS commands can also be used to animate objects and build interactive applications.

BIFS commands borrow from many concepts associated with the Virtual Reality Modeling Language (VRML). We will describe VRML in the next section of these course notes. MPEG-4 uses BIFS to support real time streaming of content. A scene does not need to be downloaded completely from the Internet before it is played. VRML uses a client-server model that relies on a complete download of 3D objects. The latest information on MPEG-4 can be found at: (http://www.cselt.it/mpeg/standards/mpeg-

<u>4/mpeg-4.htm</u>).

Reference: Rob Koenen, "MPEG-4: Multimedia for Our Time", IEEE Spectrum, Institute of Electrical and Electronic Engineers Inc. (IEEE), February 1999, pp. 26 - 33.

Stefano Battista, Franco Casalino, and Claudio Lande, "MPEG-4 A Multimedia Standard for the Third Millennium, Part 1", IEEE Multimedia, IEEE Computer Society, October/December 1999, pp. 74 - 83. See electronic version at: (http://www.computer.org/multimedia/articles/MPEG4_1.htm).



Figure # II-6: Still image from a "Streaming VRML Worlds and Mixed Media with MPEG-4" technology presentation shown at ACM SIGGRAPH 98, as part of the MAGNET Project by France Telecom R&D, France Telecom CNET/DIH. Shout Interactive Inc. created the VRML content associated with this interactive game application. The demonstration combined MPEG-4 BIFS content, Java bytecodes, and MPEG-4 audio and video, streamed over RTP. The content showed three aspects of integrated and synchronized mixed media delivery: (1) streamed, animated 3D graphics with audio; (2) point-to-point communication of audio and graphics, and (3) local 3D interaction with streamed audio and video. Advanced coding and compression technology allowed this demonstration to operate under the constraints of a 33.6. kbit /s(1) modem. More information on France Telecom R&D can be found at: (http://www.cnet.fr/). Shout Interactive Inc.'s web site is located at: (http://www.shoutinteractive.com). Image shown courtesy of Jeff Close.

II - 7: VRML/X3D - A Tool for 3-D Content and Applications on the Internet

The Virtual Reality Modeling Language (VRML) is a tool for building 3D content and applications on the Internet. It is an open specification. The Web3d Consortium is chartered to guide the evolution of VRML and the emerging Extensible 3D (X3D) standard. X3D can also be called the next-generation VRML. In February 2001, the Web3D Consortium announced the Rich Media 3D (RM3D) initiative to develop an open standard to support rich media content containing 3D graphics, video and audio over the Internet. RM3D is intended to closely interoperated with MPEG-4. We describe these developments below.

VRML is a specification originally based on an extended subset of the Silicon Graphics Inc. (SGI) OpenInventor scene description language. Introduced in late 1994 and adopted in 1995, the initial VRML 1.0 standard provided a core set of object oriented constructs augmented by hypermedia links. VRML 1.0 allowed for scene generation by Web browsers on various computing platforms, including Intel and Apple personal computers as well as UNIX workstations.

The VRML 2.0 standard, released August 1996, expanded VRML 1.0 to address real time animation issues on the Web. VRML 2.0 provided local and remote hooks (i.e. an applications programming interface or API) to graphical scene description. Dynamic scene changes are simulated by any combination of scripted actions, message passing, user commands, or behavior protocols (such as Distributed Interaction Simulation (DIS) or Java).

The VRML Consortium was established in December 1996 to foster the development and evolution of VRML. By December 1997, VRML97 was approved as the International Standard (ISO/IEC 14772) file format for describing interactive 3-D multimedia on the Internet. VRML97 is almost identical to VRML 2.0 with many editorial improvements to the standards document and a few minor functional differences

In July 1998, work began on the next-generation VRML (VRML-NG). As the Extensible Markup Language (XML), MPEG-4, and additional 3-D Web technologies began to emerge, the VRML Consortium decided to expand its charter to go beyond VRML. In December 1998, the VRML Consortium redefined itself as the Web3D Consortium. In February 1999, the Web3D Consortium expanded the focus of VRML-NG and redefined the effort as "Extensible 3D" (X3D). X3D will be a 3D standard for the Web that expresses the geometry and behavior capabilities of VRML97 using the Extensible Markup Language (XML). The first proposals for the X3D specification were available in February 1999. In August 1999, the implementation and evaluation phase of X3D began. X3D functionality was demonstrated at the Web3d-VRML 2000 Symposium, February 21 - 24, 2000 and at SIGGRAPH 2000, July 23 - 28, 2000. More information on X3D can be found at: (http://www.web3d.org/x3d.html).

Reference: George S. Carson, Richard F. Puk and Rikk Carey, "Developing the VRML 97 International Standard", IEEE Computer Graphics and Applications, IEEE Computer Society, Vol. 19, No. 2 (March/April 1999), pp. 52 - 58.

The interaction model of 3D VRML browsers is a client-server approach, similar to most other Web browsers. 3D browsers are usually embedded into 2D browsers (e.g. Netscape or Microsoft Internet Explorer) or launched as helper applications when connecting to a 3-D site. A VRML source file is down loaded from the Internet server to the Web client requesting to view the 3D VRML "world". The user interacts with this 3-D world on their own (client) workstation or personal computer. This client-server approach has limitations when it comes to multiple user interactions and content streaming with a single VRML world. Efforts are underway to incorporate VRML type capabilities into Java technology and eliminate the need for a VRML plug-in or special viewing tools.

Efforts are underway to provide streaming and multi-user capabilities in VRML and X3D. The DIS-Java-VRML Working Group of the Web3D Consortium has developed and is enhancing a public domain (freeware) Java-based implementation of the IEEE DIS protocol. Many virtual reality applications use the IEEE Distributed Interactive Simulation (DIS) Protocol for multi-user interactivity. This Working Group is also evolving a set of recommended practices for integrating DIS with VRML content and applications. For more information see: (http://www.web3d.org/WorkingGroups/vrtp/dis-java-vrml/). Another approach to collaborative VRML involves the development and construction of a networking protocol called the Virtual Reality Transfer Protocol (vrtp). Vrtp would support VRML or X3D just as the hypertext transfer protocol (http) supports HTML. More information on the Web3D Consortium's VRTP Working Group is available at: (http://www.web3d.org/WorkingGroups/vrtp/).

Announced early in 2001, the Rich Media 3D (RM3D) initiative of the Web3D Consortium is designed to facilitate an emerging category of media-capable Internet and broadcast appliances to receive and present advanced 3D content. With the intent of addressing the interactive TV market, Sony developed

much of the technology behind RM3D in its Blendo efforts. Shout Interactive has also developed pioneering 3D on the Web applications for episodic animation and wireless content. Uma, an Austrian-based technology company also contributing to RM3D, has demonstrated visually compelling solutions for visual data management and edutainment. The wide range of companies contributing to RM3D include: 3Dlabs, ATI, Eyematic, iVAST, OpenWorlds, Shout Interactive, Sony, SRI and uma. RM3D is expected to be complete with example implementation by December 2001.

Reference Web sites for Web3D Consortium, VRML/X3D and RM3D:

The Web3D Consortium,

(http://www.web3d.org/)

VRML and X3D Specification Documents,

(http://www.web3d.org/fs_specifications.htm)

Rich Media 3D Discussion.

(http://www.web3d.org/WorkingGroups/rm3d/rm3dcontacts.html)

Computer Associates' free VRML Browser - Cosmo Player,

(http://www.cosmosoftware.com/)



Figure # II-7: Still image of a VRML model of a "virtual city". This geographic visualization was originally created with ArcView 3D Analyst software, developed by Environmental Systems Research Institute (ESRI). The "virtual city" was then converted to a VRML 2.0/97 model. This work was done as part of a "Human Exposure in Urban Environments" project for the United States Environmental Protection Agency (US EPA) - Alan Huber, primary investigator. The actual 3D Web environment can be found at: (http://www.epa.gov/gisvis/arcview/). You need a VRML plug-in for your Web browser to view this 3D World. Image created by Theresa-Marie Rhyne, while working for Lockheed Martin Technical Services at the US EPA Scientific Visualization Center.

II - 8: Proprietary Web3D Format: 3DML

There are a number of proprietary formats for creating Web3D solutions. Cult3D, a plug-in to Adobe Acrobat, from Cycore is an example, <u>see: (http://www.cycore.com/)</u>. For the remainder of this section of these course notes, we will focus on an easy to learn proprietary XML type language entitled the Three Dimensional Markup Language (3DML), <u>(see:</u>

(http://www.xml.com/xml/pub/1999/01/3dml/3dml.html).

3DML was released in late December 1998, by Flatland Online Inc., as a markup language for rapidly and easily creating 3-D Web sites, (see: (http://www.flatland.com/enter.html). Flatland provides a freely available and downloadable browser extension, entitled "Flatland Rover". The Flatland Rover Web browser extension displays content created with 3DML. Flatland Rover works on Windows 95/98 and Windows NT 4.0 platforms. 3DML also supports streaming audio and video as well as live video and spatialized audio.

The Flatland Web site provides a step by step online tutorial to learning 3DML, (http://www.flatland.com/build_direct.html), entitled "3DML Builder's Guide. Content created with 3DML is called a "SPOT". 3D Spots created with 3DML fundamentally differ from virtual "Worlds" created with the Virtual Reality Modeling Language (VRML). The difference is that 3DML Spaces are built without having to do any 3D modeling. In 3DML, there are block sets that are used to build a 3DML virtual space. The "basic" 3DML block set contains objects that are the shape of ramps, columns, signs, and other basic primitive objects. The "village" 3DML block set contains blocks shaped like trees, houses and sidewalks. In addition, there are advanced block sets for creating complicated Spots. Since no modeling is required, 3DML Spots are easily authored with a text editor.

Like an HTML file, a 3DML file has two primary sections: the <HEAD> and the <BODY>. The <HEAD> is the section where parameters are set that are true for the entire Spot. This includes the map dimensions, ambient light, sound, and the sky texture. The <BODY> is the section where blocks are customized and where the actual map of the 3D space is defined. The map is the heart of a 3DML file, where different kinds of blocks (represented by ASCII characters) are used to define a 3D space. Each block occupies a space of 256 x 256 x 256 (pixels). Blocks are arranged into a grid pattern to create each horizontal level of a Spot. Levels are stacked on top of each other to create multi-story Spots.

Below is an example of how to define a map that will create a "one level building and its entrance" using 3DML.

```
<LEVEL NUMBER="1">
###...##
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
#....#
```

3DML supports streaming audio and video in either Window Media (.avi) and/or Real Media (.rm) format. Presently, Flatland's Rover brower does not support other streaming media formats or SMIL multi-media presentations. Flatland provides an excellent tutorial on how to incorporate streaming media into 3DML, see: (http://www.flatland.com/build/tutorial/adv_streaming.html).

Many 3DML examples tend to be oriented towards the Computer Games, Live Concerts or E-commerce arenas. Below, we show an image taken from a 3DML example built from our own texture maps. We used Adobe PhotoShop for creating our 128 by 128 pixel images that were saved as GIF files. We found

this GIF size produced the least number of 3DML coding errors.

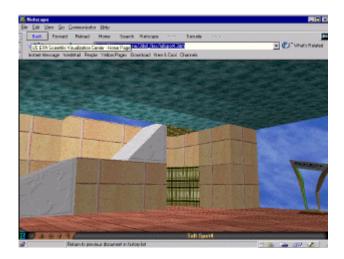


Figure #II - 8: Still image from a 3DML "Spot" created with customized texture maps. Similar to XML, 3DML is a markup language for creating 3-D Web content. Flatland Online Inc. developed 3DML and provides a free downloadable brower for viewing 3DML files. More information on 3DML can be found at: (http://www.flatland.com/enter.html). Image and 3DML Spot created by Theresa-Marie Rhyne. Special thanks to Christopher Nash, at Flatland Online Inc., for providing helpful guidance in building 3DML Spots.

<u>Click Here to View our example 3DML Spot from your Web Browser</u>. The Flatland browser (Rover) must be downloaded to your machine to view these 3DML files. Flatland's Rover only works on Windows 95/98/NT platforms.

II - 9: Brief Overview of Java3D and DirectX

Two additional technologies that allow for building 3D Internetworked applications are Java3D from Sun Microsystems Inc. and DirectX from Microsoft Corporation. A brief mention of these technologies is provided here.

Java 3D is an application programming interface (API) for writing networked 3-D applications and 3-D applets with Java. Like VRML, Java 3D uses the scene graph paradigm. A scene graph is a hierarchical representation of objects and behaviors for the virtual "world" or application you are building. Java 3D's geometry objects contain the actual geometry to be rendered in 3-D scene. Utility packages bundled with Java 3D include: (a) content loaders; (b) scene-graph construction aids; (c) geometry classes; and (d) convenience classes. Convenience classes include techniques for mapping textures on to objects. More information on Java 3D can be found at: http://java.sun.com/products/java-media/3D/). An example of a Java3D application is provided in "Section II - 9: Interactive Visual Applications & Cooperative Work: Example #2: VisAD - A JDK 1.2 & Java 3D Collaborative Application" of these course notes.

Microsoft's **DirectX** is a set of APIs and programming tools for creating Windows-based applications and digital content that includes full-color graphics, video, 3D animation, and surrounding sound. DirectX is incorporated into Windows 98, Windows 2000 and Microsoft Internet Explorer. DirectX capabilities for Windows 95 can also be downloaded from the Microsoft Web site. More information on DirectX is available at: (http://www.microsoft.com/directx/homeuser/aboutdx.asp).

II - 10: Examples of Visual Applications & Cooperative Work

Many projects around the world are examining methodologies to support Intenetworked 3D graphics applications, related human factors issues, and their economics and benefits for specific application domains. We describe two examples below that address multi-user interaction within a single application.

II - 10a: Example #1: The Multicast internet Maze (MiMaze) & MiMaze3D in VRML

The Multicast internet Maze (MiMaze) is a pioneering distributed computer game, initially developed in 1997, that uses the multicast backbone (MBone). It was designed by researchers in the Network and Systems group at the French National Institute for Research in Computer Science and Control (INRIA) in Sophia Antipolis, France. In this real-time interactive game, players battle against one another while moving through a virtual (maze) environment. MiMaze is a distributed network application on the Internet with no file server. Participants in the game exchange information about their position or location using the Distributed Interactive Simulation (DIS) protocol set of rules. MiMaze3D is the VRML extension of MiMaze. Real time navigation is accomplished, in MiMaze 3D, with VRML's External Authoring Interface (EAI). More information on MiMaze can be found at: (http://www.inria.fr/rodeo/MiMaze/)



Figure # II - 10a: Example session from the MiMaze3D game. This distributed computer game uses the multicast backbone (MBone) and VRML technology. Image courtesy of Christophe Diot, the French National Institute for Research in Computer Science and Control (INRIA) in Sophia Antipolis, France. See: (http://www.inria.fr/rodeo/MiMaze/MiMaze3D.html).

Reference: Laurent Gautier and Christophe Diot, "Design and Evaluation of MiMaze, a Multi-player Game on the Internet", IEEE Multimedia Systems Conference, Austin, Texas (USA), June 28 - July 1, 1998. (http://www.inria.fr/rodeo/MiMaze/ReportInt.html).

II - 10b: Example #2: VisAD - A JDK 1.2 & Java 3D Collaborative Application

The Visualization for Algorithm Development (VisAD) system was designed, in 1992, to allow scientists to use interactive visualization for examining their computational algorithms and models. The first version of VisAD was developed at the University of Wisconsin at Madison and used its own unique programming language on Silicon Graphics workstations.

In 1997, the developers of VisAD re-introduced their system as a class library in the Java programming language. This new version of VisAD requires JDK 1.2 and Java 3D. Since VisAD is written in pure Java, the application is platform independent. VisAD supports collaborative work by allowing users at multiple locations to share control over a single user interface to the interactive visualizations. More information on the new version of VisAD can be found at: (http://www.ssec.wisc.edu/~billh/visad.html).

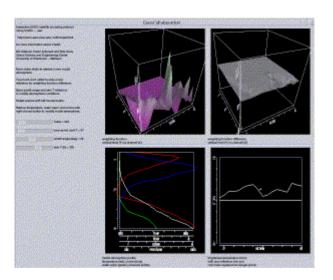


Figure II - 10b: A collaborative application of GOES satellite sounding data built with VisAD - a JDK 1.2 & Java 3D system. Image courtesy of Bill Hibbard, University of Wisconsin at Madison. See: (http://www.ssec.wisc.edu/~billh/vis.html).

Reference: William Hibbard, John Anderson and Brian Paul, "A Java and the World Wide Web Implementation of VisAD", Conference on Interactive Information and Processing Systems for Meteorology - 1997, American Meteorology Society, August 1997. Available at: (http://www.ssec.wisc.edu/~billh/amsvisad.txt).

II - 11: Concluding Remarks: Integrated Internetworked 3D Graphics

The future development of Web software and applications requires effective integration of the Internetworked 3D Graphics concepts described in these notes. The rapid development of the Internet and the Web has already begun to foster collaborative information discovery and visual exploration. Multicasting, the Real Time Streaming Protocol (RTSP), and the Wireless Application Protocol (WPA) connectivity tools provide the foundation to support streaming multimedia via the Internet. As a result, Synchronized Multimedia Integration Language (SMIL) applications are evolving and expanding. MPEG-4, 3DML, VRML/X3D, and the emerging RM3D standard provide guidelines for delivering 3D content. Many applications are being rewritten or initially created with Java and Java3D or DirectX to optimize interoperability across the Internet. The VisAD system, described in these notes, is just one example of such efforts. With the multicast backbone (MBone), the telecommunications community has already provided prototype desktop videoconferencing capabilities. Multicasting techniques have even begun to be applied to 3D applications like the MiMaze3D example highlighted in these notes. Many Working Groups of the Web3D Consortium have stepped beyond the client-server interaction model to address streaming, multi-user interaction, and large-scale virtual worlds. Clearly, we have only begun to tap the potential of Internetworked 3D Graphics and to understand its capabilities to deliver 3-D interactive content to the general community of users.

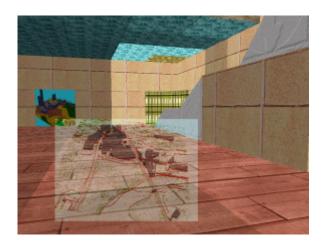


Figure II - 11: Artistic Web 3D composition, entitled "Standing on Virtual Ground", intended to depict future content for Integrated Internetworked 3-D Graphics. Created by Theresa-Marie Rhyne, 1999.

II - 12: Acknowledgements

These notes and most of the images are the result of many conversations and insights from my colleagues in computer graphics and networking. A warm thank you to: Tamara Munzner, Jeff Brown, Hans-Werner Braun, Peter Alesso, Andy Smith, Jeff Close, Alan Huber, Christopher Nash, Christophe Diot, Bill Hibbard, and Martin Reddy. Additional thanks to Bob Barton, Don Brutzman, Grantley Day, and Mike Macedonia for thinking an Internetworked 3D Graphics course was a good idea and for collaborating with me on this instructional effort. Final thanks to Lou Harrison, ACM SIGGRAPH 2001 Courses Chair, and the SIGGRAPH 2001 Courses Committee for accepting our proposed course vision.

Overview of 3D Interactive Graphics Using the Internet

Theresa Marie Rhyne Independent Consultant 3D Computer Graphics Technologies rhyne@siggraph.org

Defining Internetworked 3D Graphics

The convergence of networking and 3D computer graphics to support collaborative exploration and visualization on the Internet and the World Wide Web.

Six Components of Internetworked Graphics

- Connectivity
- Economics

- Content
- Applications
- Interaction
- Personal Impacts

MBone: The Multicast Backbone for Live Video and Audio on the Internet

- High Bandwidth Internet Backbone Connections
- Desktop Computers with built-in video & audio
- Public Domain Software Tools

Examining the Internet Protocol (IP)

- Unicasting: the client-server model
- Multicasting: one to many model
- Real Time Streaming Protocol RTSP
- Wireless Application Protocol WAP

Rich Media Across the Internet

- RealNetworks: http://www.realnetworks.com/
- Apple QuickTime:

http://www.apple.com/quicktime/

• Microsoft - Windows Media Technologies:

http://www.microsoft.com/windows/windowsmedia/

• Viewpoint: http://www.viewpoint.com/

Streaming Media over the Internet

The Concept: It is not necessary to completely download streaming content before it is played.

SMIL - the Synchronized Multimedia Integration

Language

Viewpoint - 3D Objects over the Internet

MPEG: Motion Pictures Experts Group - ISO Standards

- MPEG-1: digital video player
- MPEG-2: digital tv signal compression
- MPEG-4: Multimedia streaming content
- MPEG-7: digital information searching
- MPEG-21: Multimedia Framework

The Virtual Reality Modeling Language (VRML):

A Tool for Building 3D Web content

A Brief History of VRML

- VRML 1.0 extended subset of SGI's Open Inventor
- VRML 2.0 addressed real time animation on the Web
- VRML 97 approved ISO Standard
- X3D next generation VRML compatible with XML

Rich Media 3D (RM3D):

- announced February 2001
- Open Standard: streaming 3D graphics, video & audio
- Integrated with MPEG-4
- based on Sony's Blendo efforts
- incorporates Shout Interactive & uma technologies
- expected release in December 2001

The Extensible Markup Language (XML) & 3DML

- XML next generation web development
- 3DML new 3D Markup language like XML
- 3DML Spots are easy to build see example

Brief Overview of Java3D & DirectX

- Java3D programming 3D Web applications
- DirectX Microsoft's digital content APIs

Interactive Visual Applications & Cooperative Work

Providing real time collaborative information discovery and visual exploration among geographically remote users.

2 Examples of Visual Applications & Cooperative Work

- MiMaze 3D developed at INRIA, France
- VisAD (Java) developed at Univ. of Wisconsin

MiMaze 3D

- Real Time VRML 97 navigation
- Optimal Applet-Application Interaction
- Network and Graphics event scheduling
- MPEG-1 & MPEG-4 texture mapping
- Spatial audio

VisAD: A JDK 1.2 & Java3D Collaborative Application

- Original System: own programming language
- Rewritten as class library in Java 1997
- Multiple locations share control of application
- Supports heterogenous platforms

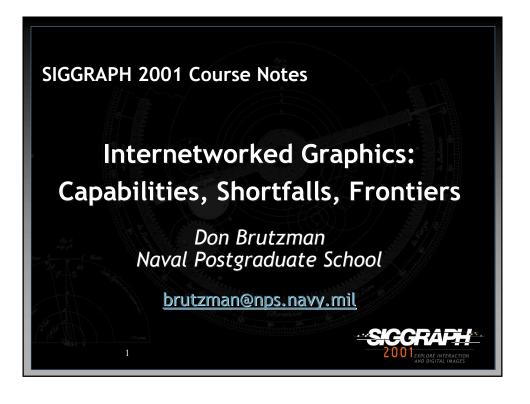
The Future: Integrated Internetworked 3D Graphics

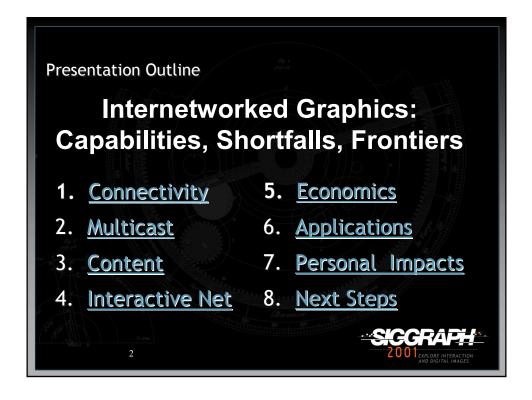
delivering 3D interactive content to the general community of users

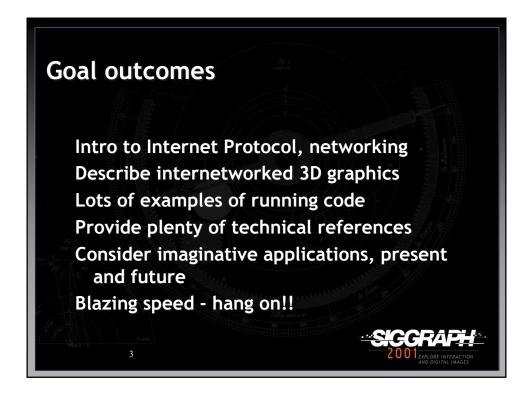
The Realities of Implementing Internetworked 3D Graphics: Computer Games Case Study

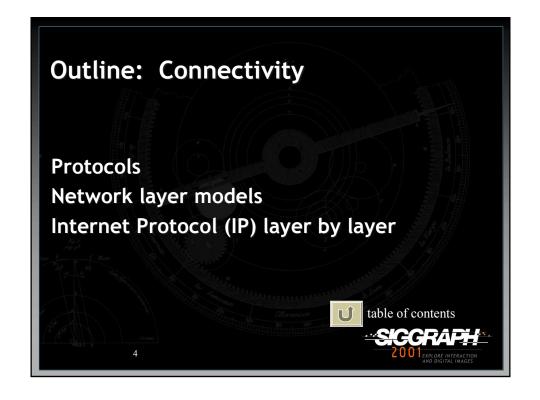
Grantley Day Maxis/Electronic Arts Gday@maxis.com

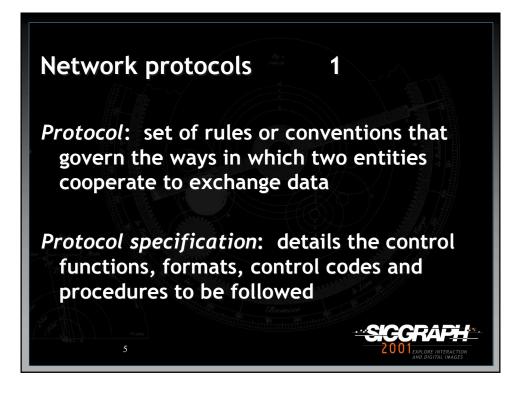
These notes will be distributed as handouts at SIGGRAPH 2001.

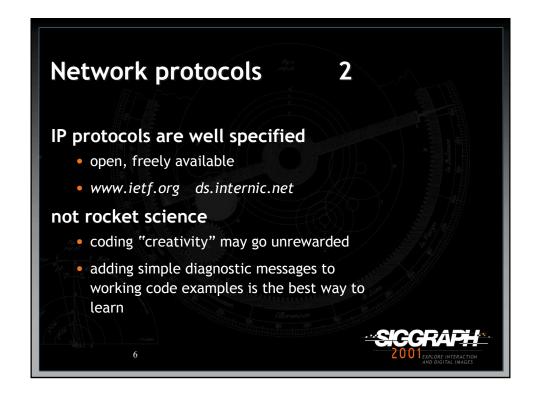


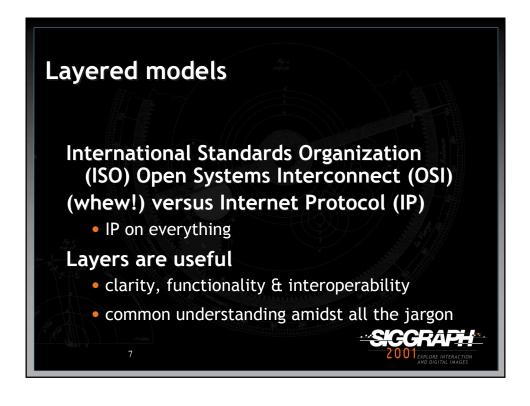


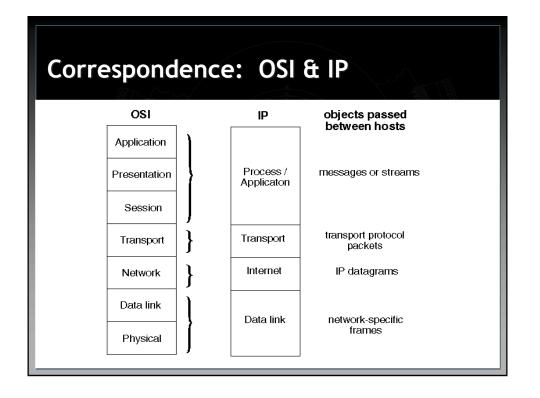


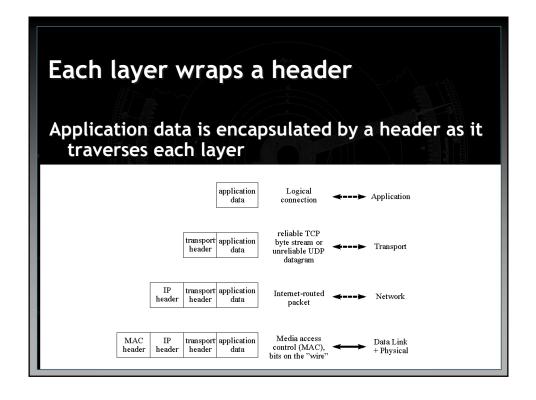


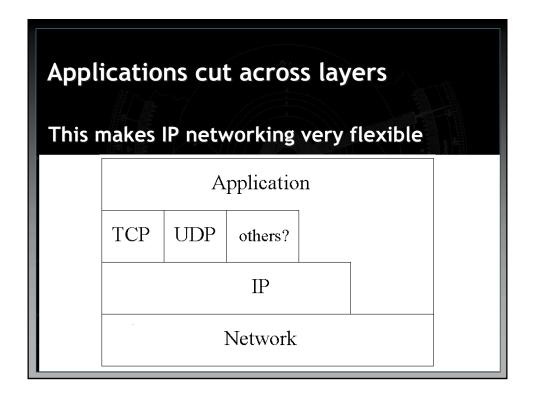














Application layer attributes

Application layer communications are diverse User applications invoke TCP/IP services to send/receive data

Single messages or lengthy streams with one or multiple other hosts

Delivery intermittent or continuous

Usually application-layer interface is embedded inside the application



Application Layer interfaces

Sockets

• one-one or many-many, reliable or unreliable

Uniform Resource Locators (URLs)

• query servers, access scripts & databases

Various virtual world toolkit libraries

- no single winner, your mileage may vary
- Kent Watsen working on Bamboo plugin architecture <u>http://www.npsnet.org/~watsen/Bamboo</u>

-SCCRAP

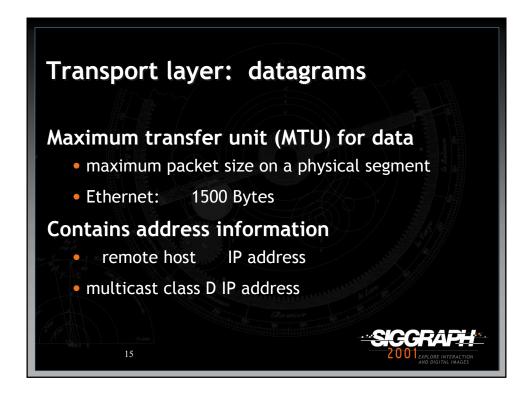
11

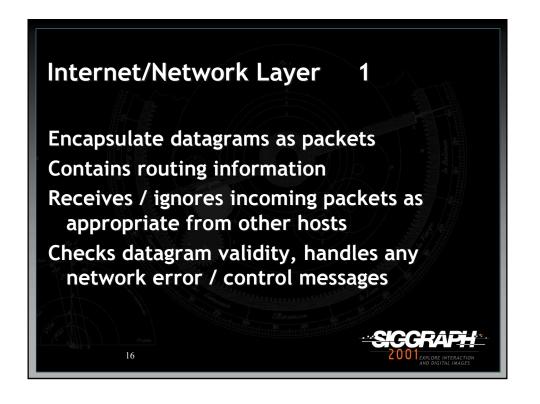
Transport Layer

Host-host byte stream or datagram communication between applications

- reliable delivery connection-oriented TCP
- unreliable delivery connectionless UDP
- experimental "reliable multicast" for trading off reliability, speed, overhead of packet delivery

Goal: exchange datagrams end-to-end with other hosts





Internet/Network Layer 2

Router: a network node connected to two or more networks that forwards packets from one network to another

Routing: process by which nodes exchange topological information to build correct forwarding tables

17



Data Link/Physical Layer 1

Includes physical media signaling and lowest-level hardware functions

Examples: network interface card together with twisted pair wire, coaxial cable or fiber



Data Link/Physical Layer 2

Exchanges network-specific data frames with other similar devices

Includes capability to screen multicast packets by address/port numbers at the hardware level

• Example: listening to all conversations at once doesn't scale well

-SICGRAPH

--SIGGRAPH

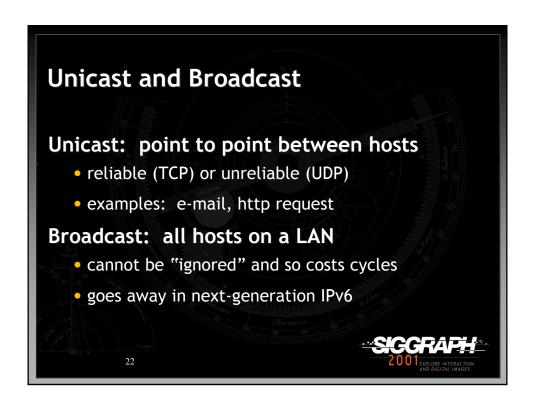
19

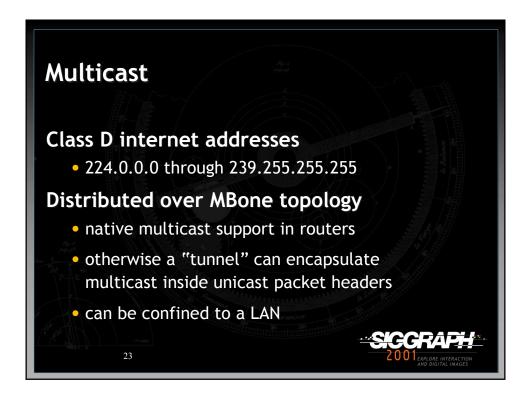
Physical layer comments

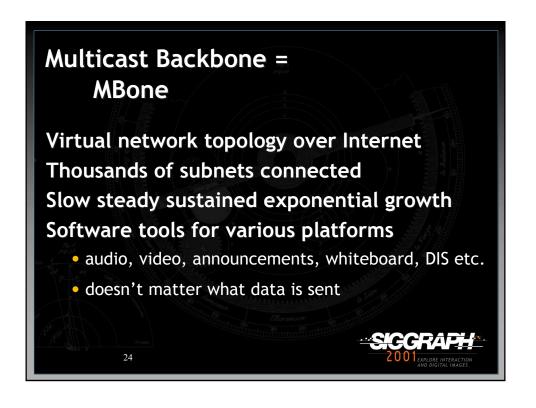
Physics of underlying media can impose significant differences in signaling ability

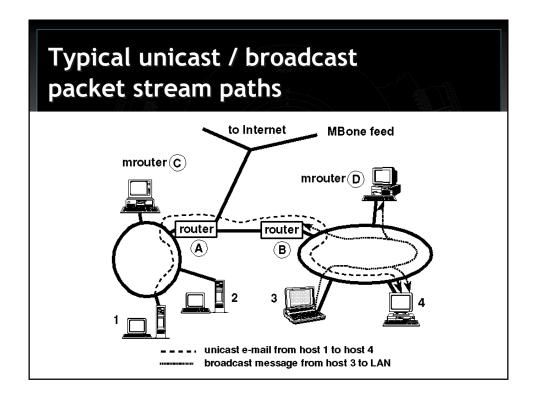
Strength of Internet Protocol is that these disparate physical connections can all interoperate, despite huge differences in links

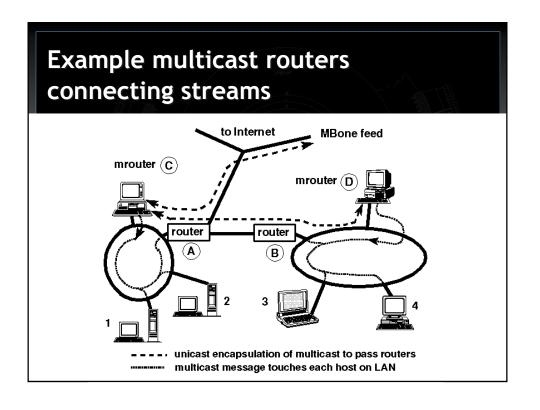


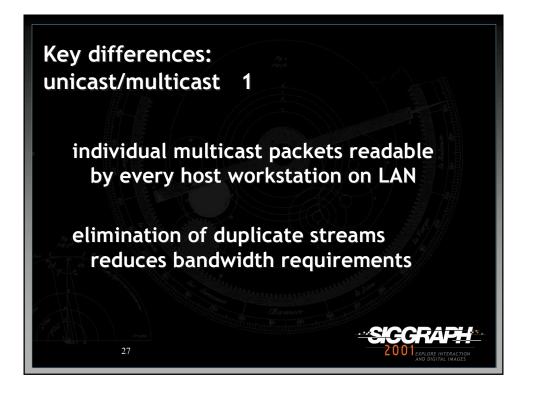


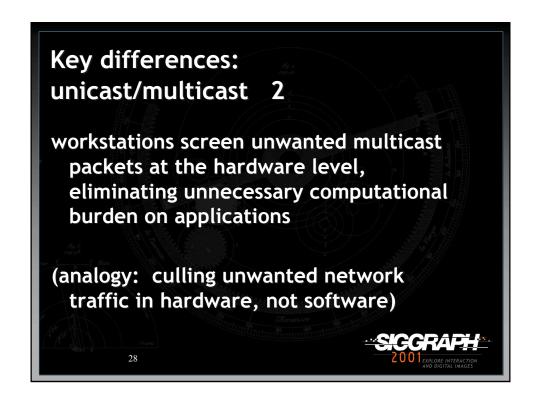












Bandwidth constraints

Frame rate versus bandwidth

human perception of smooth motion 7-8 Hz

Global MBone historically 500 Kbps

ADSL, T1: 0.5-1.5 Mbps typical offsite link

Ethernet: 10 or 100 Mbps

time-to-live (ttl) field limits data travel

administrative scope boundaries are also

possible (e.g. campus restrictions)

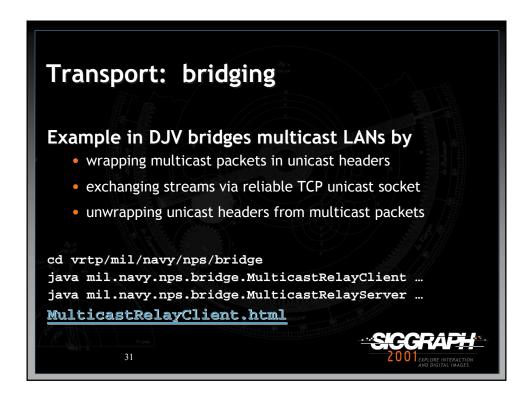
29

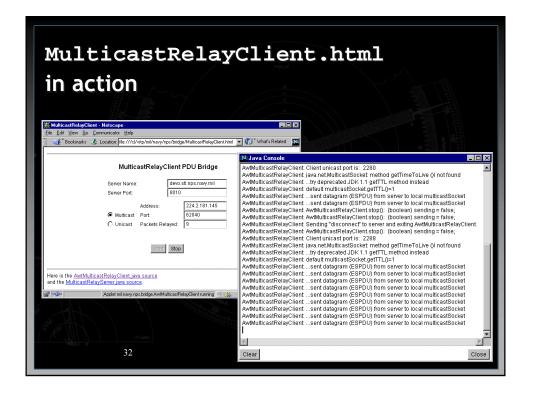
Transport: reliable multicast

Many variations in use, development

Selective reliable transport protocol (SRTP) by Mark Pullen at George Mason University (GMU), Fairfax Virginia USA http://netlab.gmu.edu/srtp

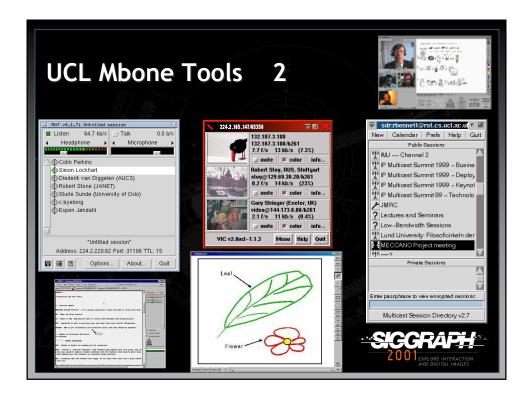
SCCRAPITY 2001 EXPLORE INTERACTION

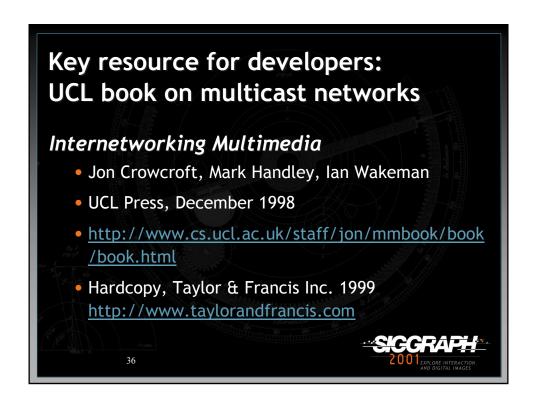






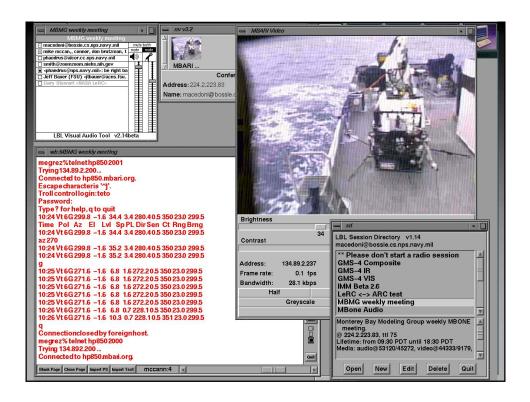


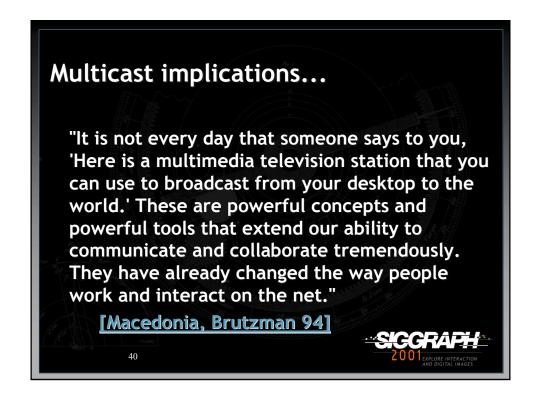
















World Wide Web Internet is our global supercomputer Web is our global super-database Sustained exponential growth rates • ~10% per month hardware (Moore's Law) • No longer easy to measure: • ~15% per month Internet hosts • ~20% per month content on the Web All media, machines, people - of course!

43

VRML Overview 1 Virtual Reality Modeling Language Began in earnest at SIGGRAPH 94 VRML 97 is ISO-approved version of 2.0 VRML 200x (X3D) is next-generation version 3D Web browsers freely available for (most) all computing platforms Now gaining X3D, open/community source Web-wide dynamic 3D graphics

VRML Overview 2

Large-scale internetworked 3D worlds are now possible

Scaling up is both a technical challenge and a people challenge

Loadable/storable via many apps, APIs

Interchange format: "Content is King"

VRML can do most jobs, as partner or alone

-SIGGRAPH

45

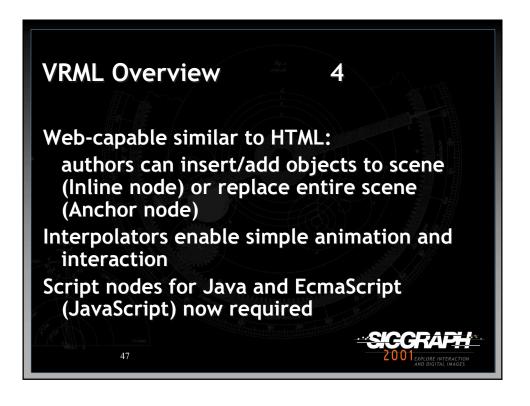
VRML Overview 3

Dynamic 3D scene description language with behaviors and animation
Rich set of graphics primitives gives

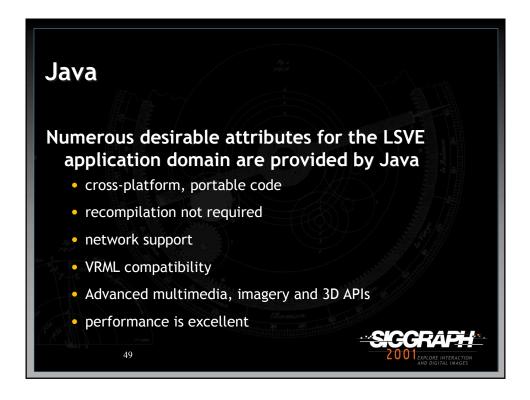
common-denominator file format
Well documented, plenty of examples

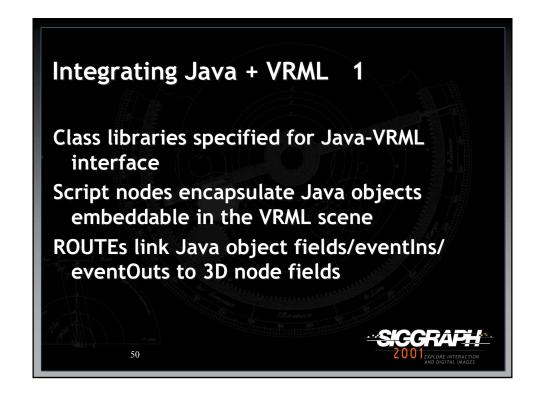
Scene graph compatibility with variety of compiled code → every API can "win"

SICGRAPIII
2001 ENRI DE INTERACTION









Integrating Java + VRML 2

Java objects can query (and interact with) browser as well as Web

createVRMLFromString method enables Java objects to generate arbitrary VRML geometry

51



Integrating Java + VRML 3

See Annex C of VRML Specification for details

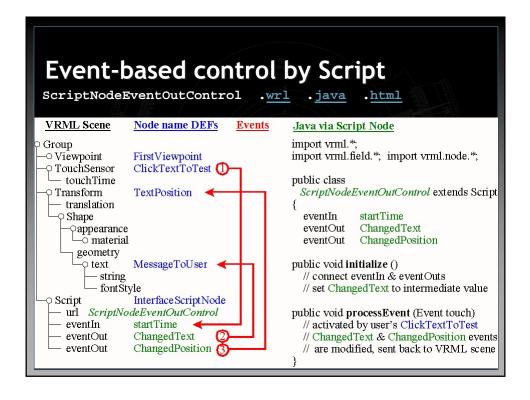
Java Script Authoring Interface (JSAI)

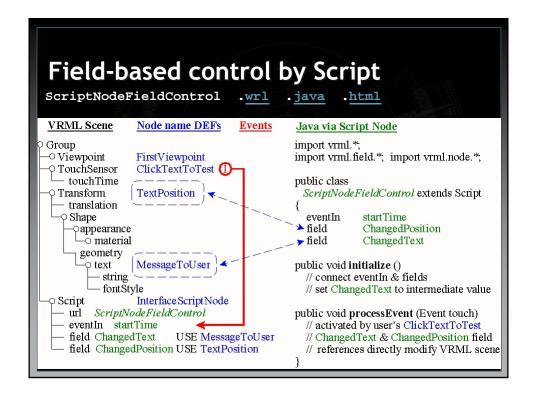
External Authoring Interface (EAI) is now a revised/implemented specification providing alternate Java-VRML-browser methodology

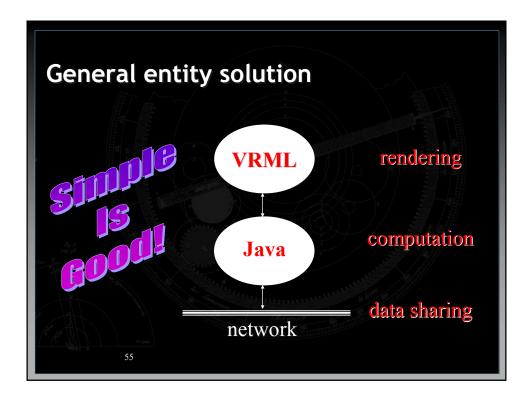
Event = changing a value in the scene graph

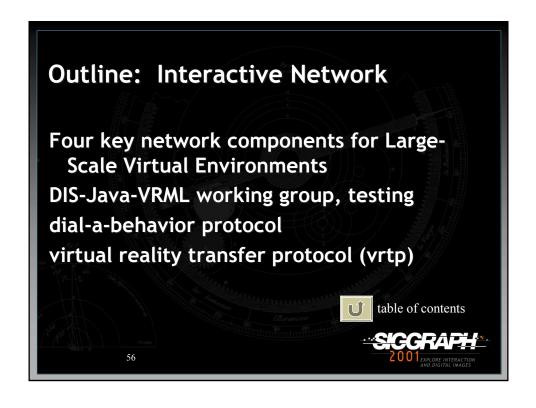
See Brutzman reference paper on VRML + Java

SIGGRAPI









Four key network components for LSVEs

Light-weight interactions

Short state/event messages via multicast

Network pointers

• URLs to information resources

Heavy-weight objects

reliable client-server response

Real-time streams - audio, video etc.

57

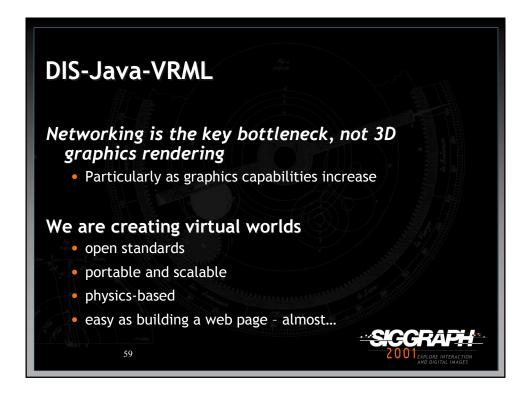
IEEE standard DIS protocol

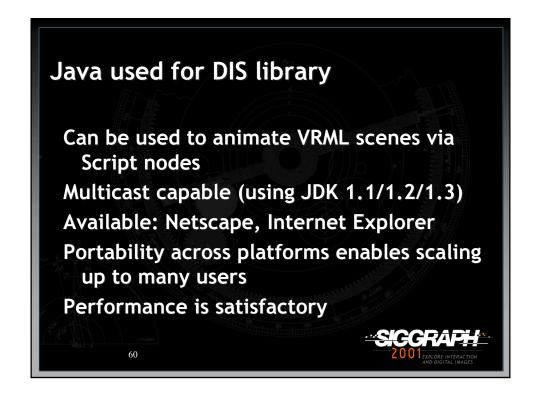
Large-scale multicast applications
Truly many-many communications
Single entity bandwidth ranges from
230..15,000 bps (bits per second)

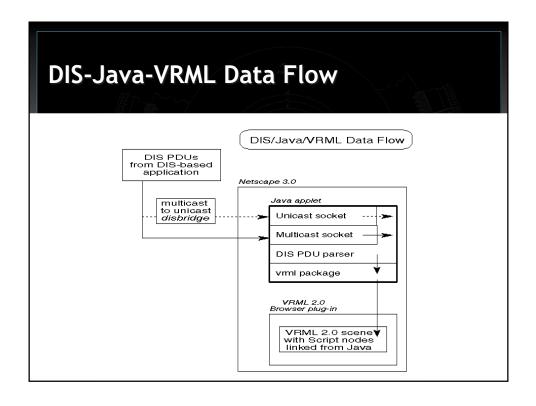
Latency & jitter compensated by timestamps & dead reckoning

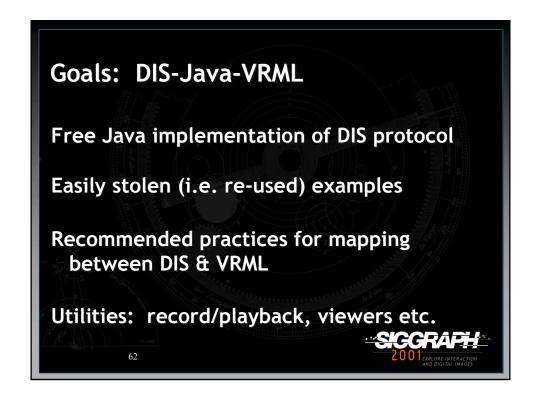
Example behavior/interaction protocol

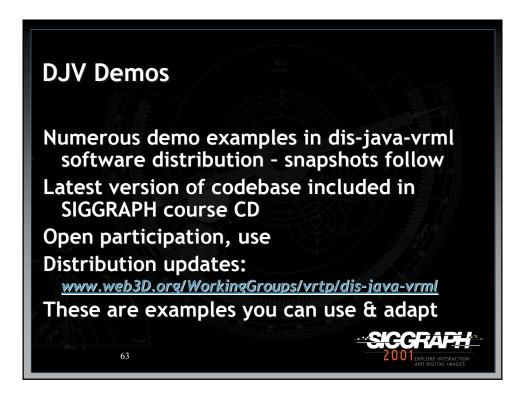
SECTAPITY 2001 EXPLORE INTERACTION



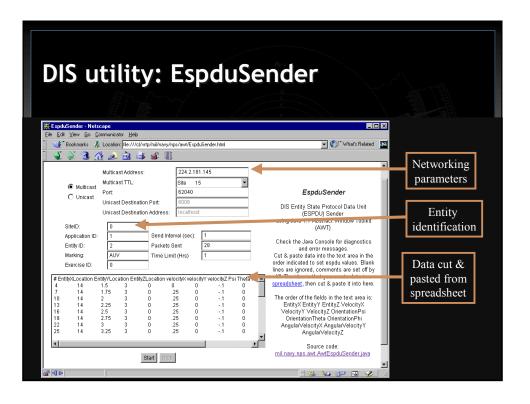


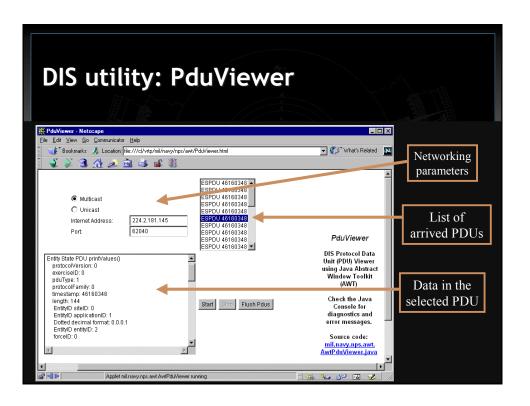


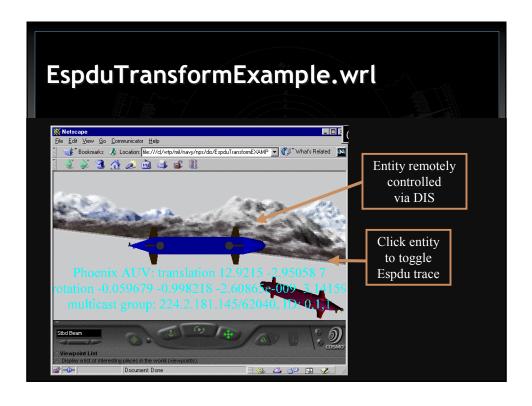


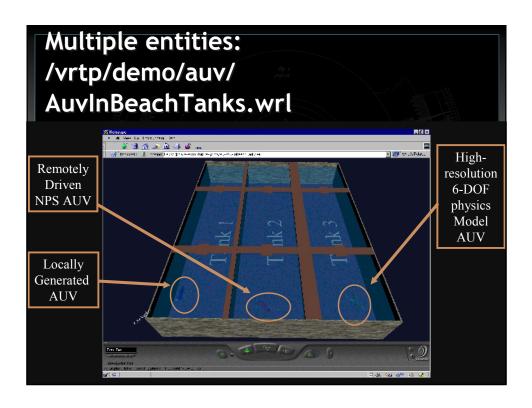


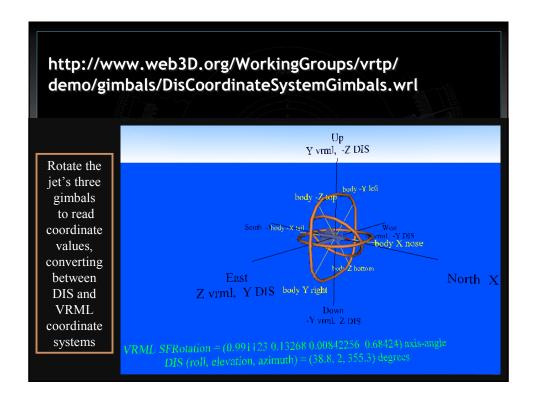


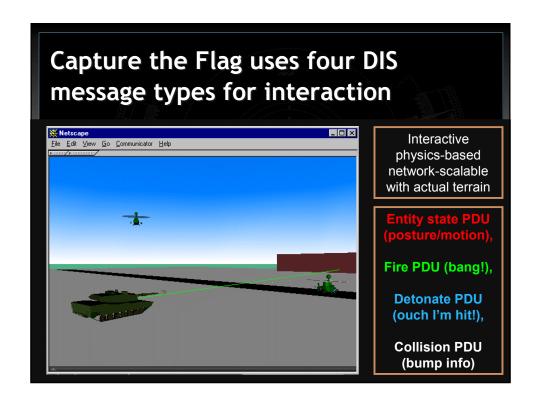


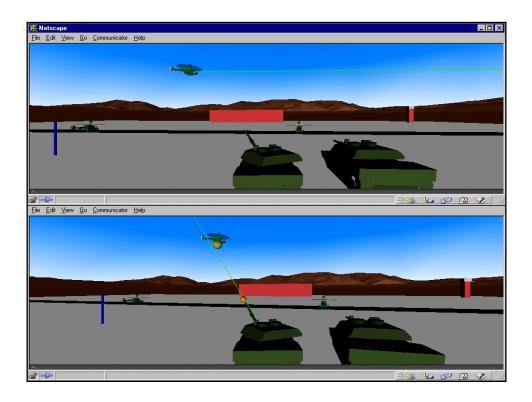




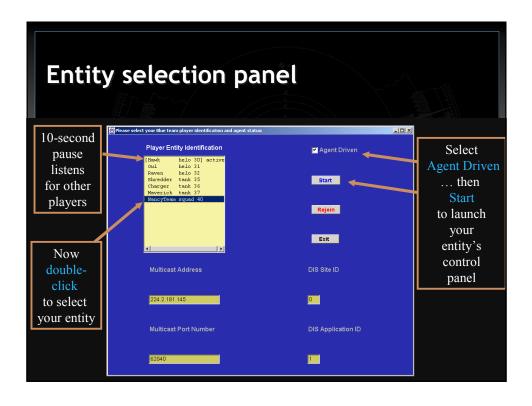


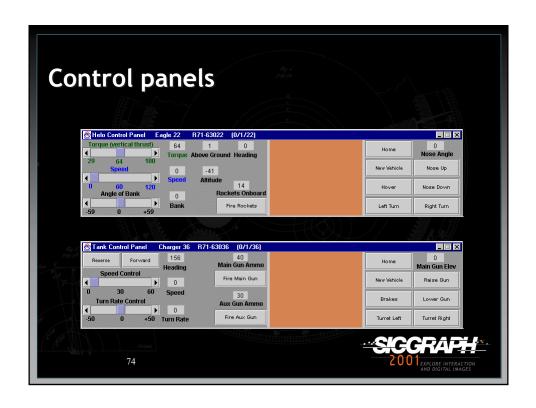




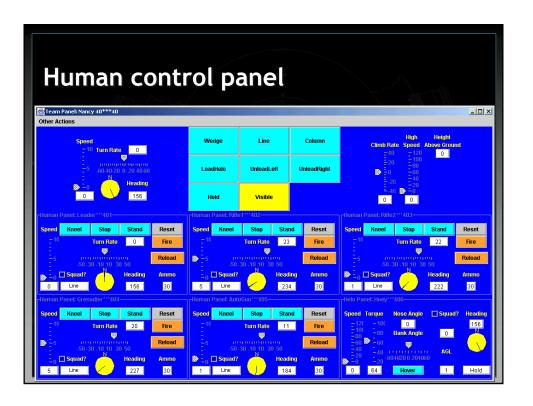






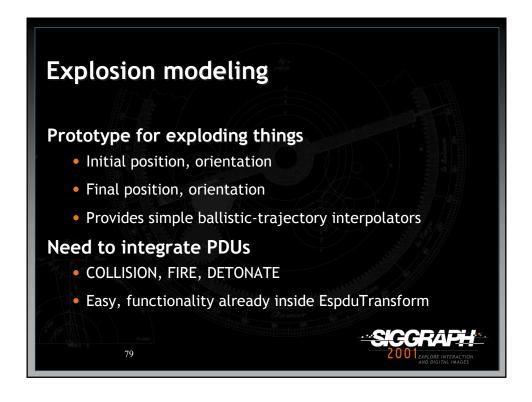


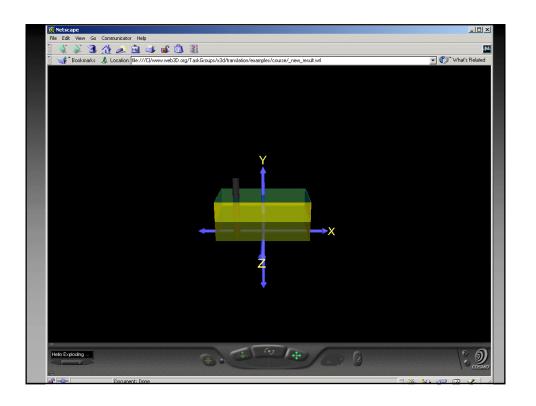


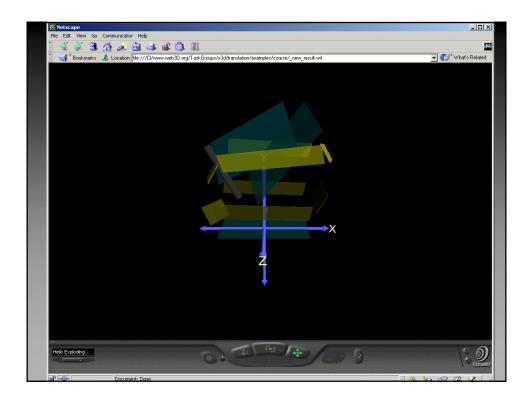


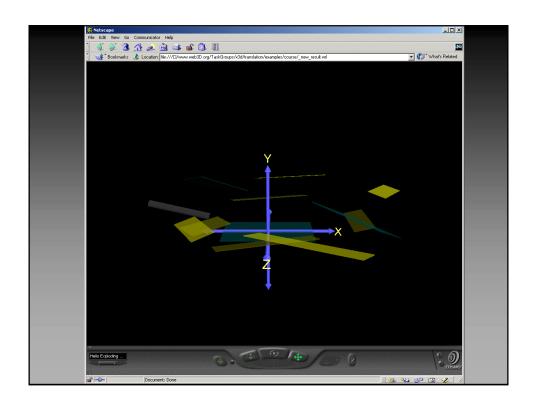


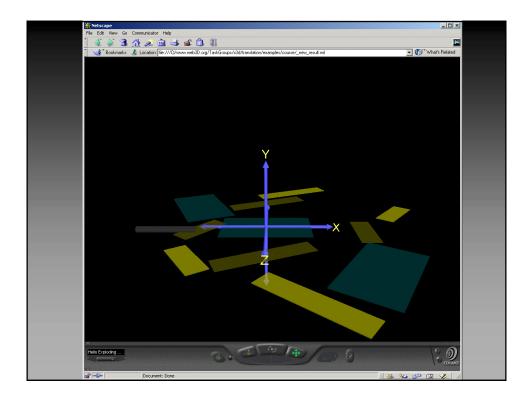




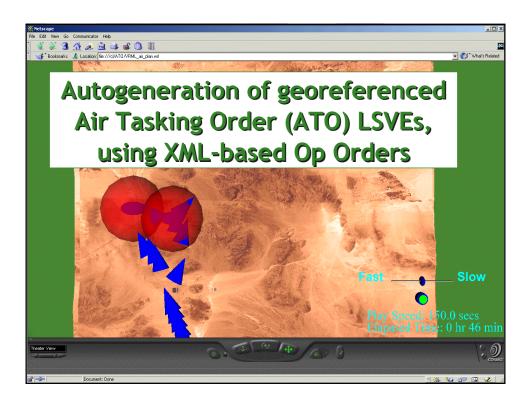


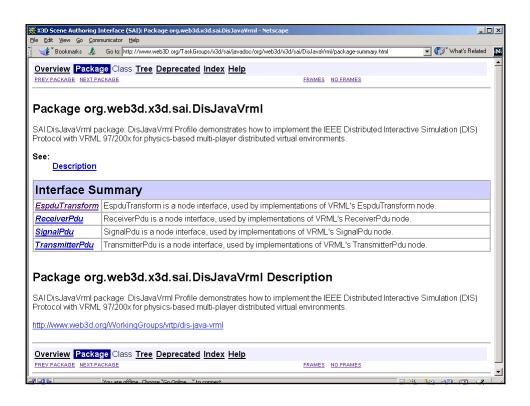


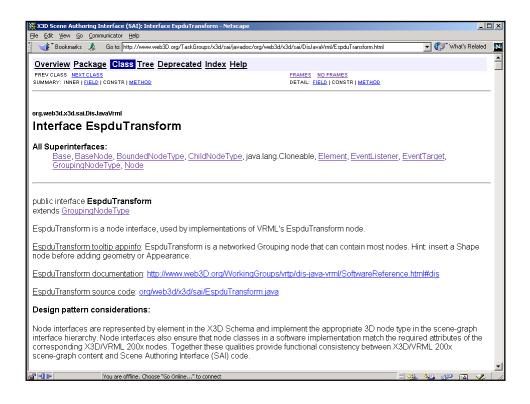


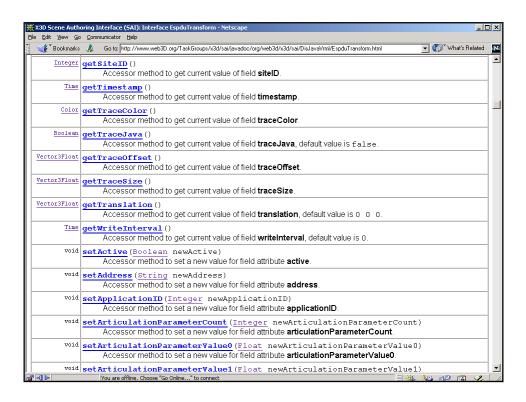




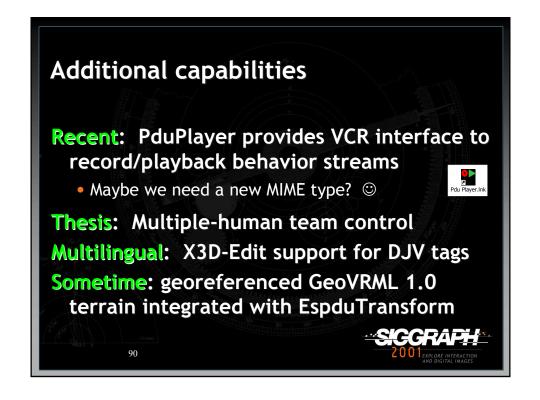


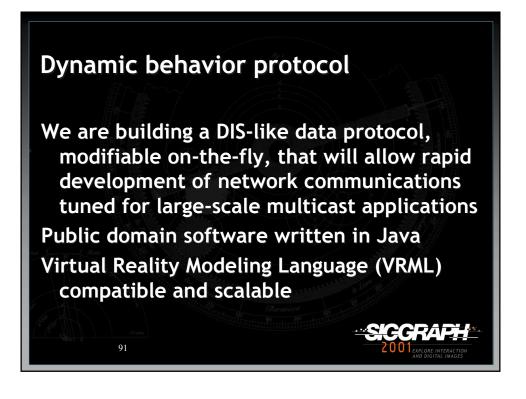


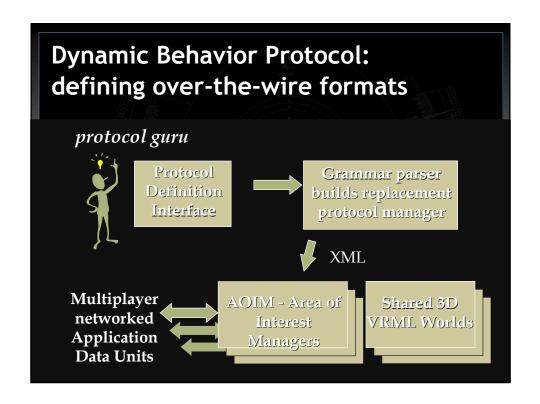


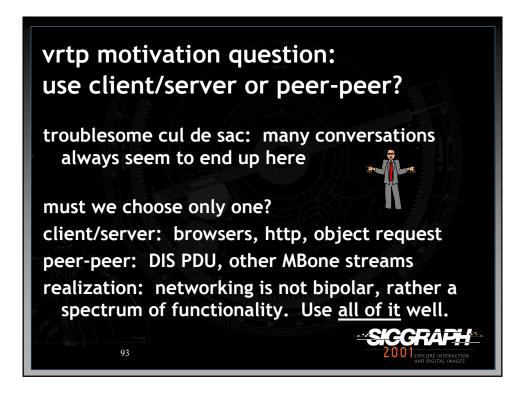


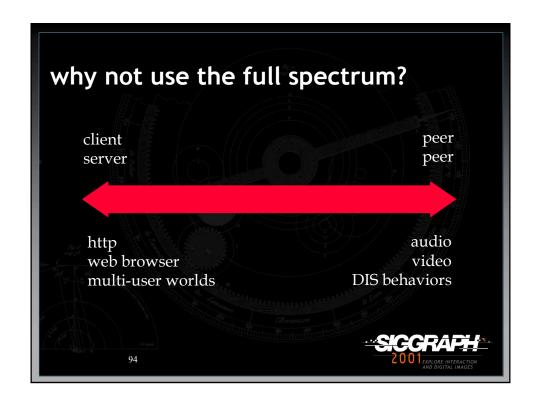
Build testbed lab at NPS and GMU, elsewhere • George Mason University (GMU) • various masters and Ph.D. students Goal: around-the-clock exercises • statistics, toolset development • Low bandwidth global on MBone Emphasis on end-user results Other sites welcome to collaborate

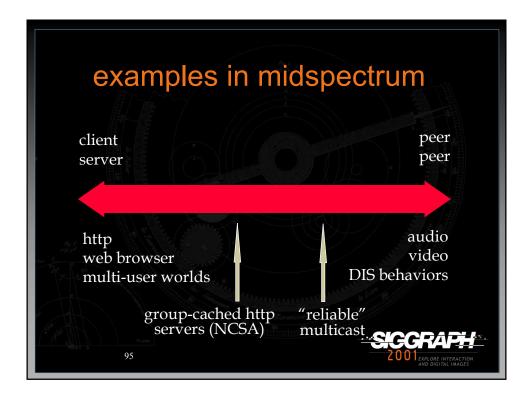












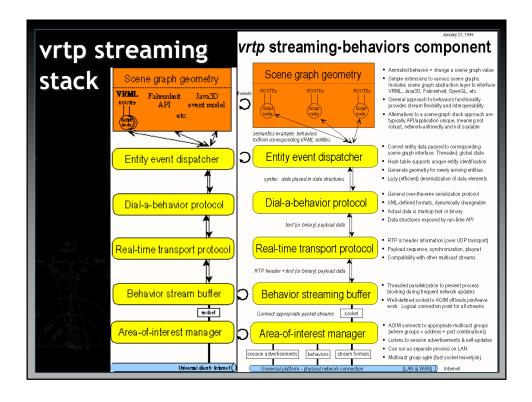


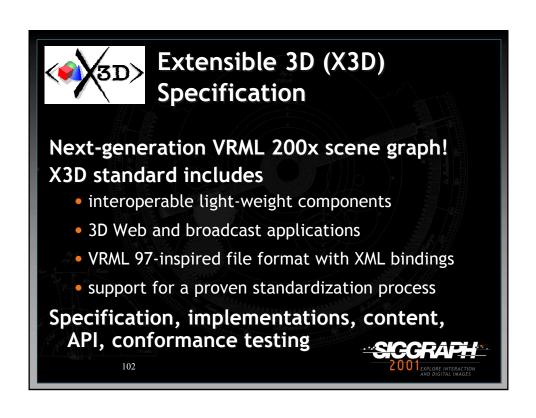


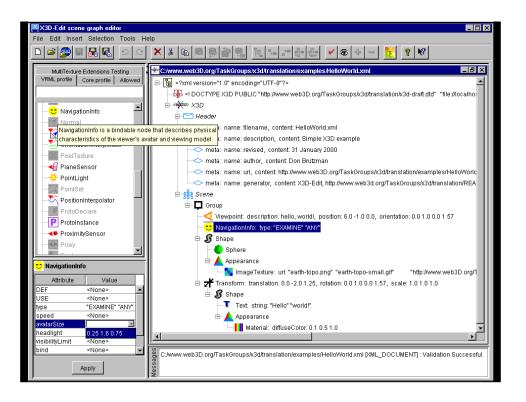


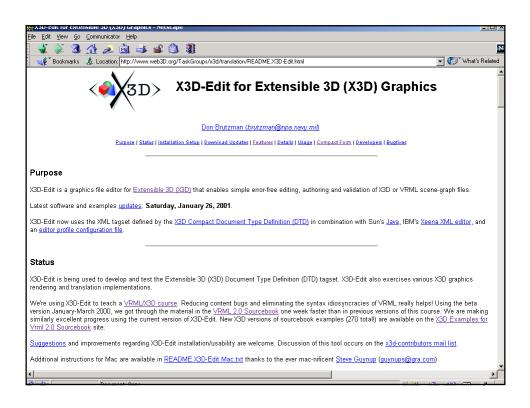


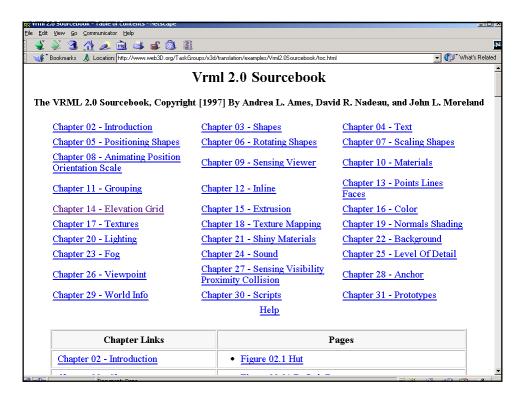


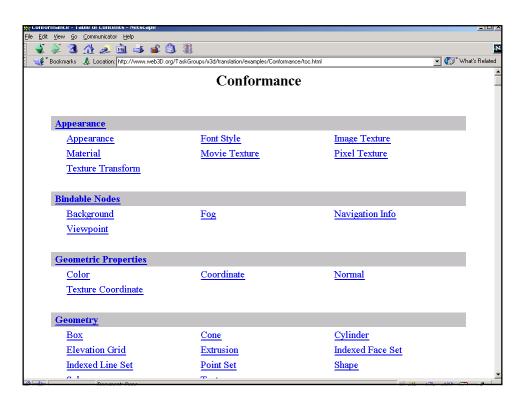












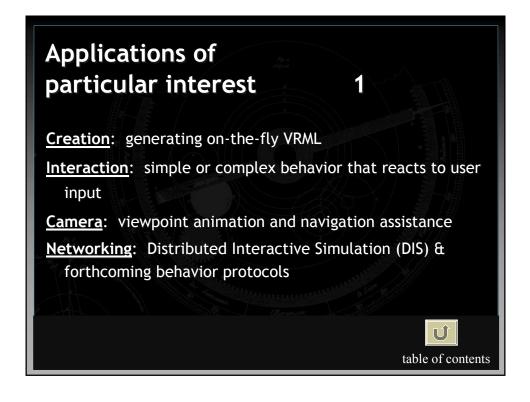


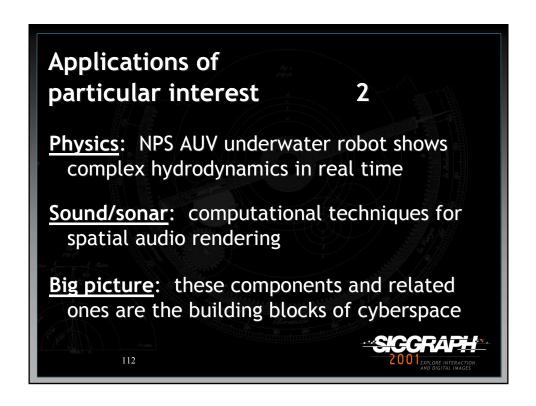


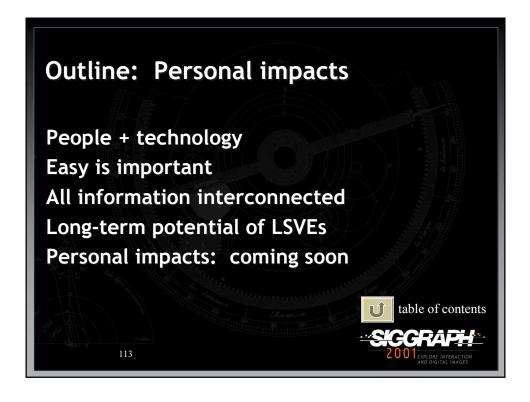
Economics issues 2 Information dissemination can be completely decentralized Open standards must grow and adapt Closed/commercial standards wither & die Slow semi-open standards fall behind and effectively appear closed

109

Economics issues Security issues (authentication, encryption, signatures, watermarks, nonrepudiation etc.) Intentionally obfuscated by Sun, Microsoft, Netscape (although see dis-java-vrml cross-browser solution) Intentionally difficult to produce: market for Verisign et al. Interesting: all issues meet in avatar technology Misunderstanding/misinformation gets worse Best approach: implement, test, evaluate









Easy is important

The right answer has to be very easy, or people won't bother doing it

"Easy as building a web page" is a meaningful metric for easy enough

Until networked graphics are that easy, shared-3D growth won't scale with Web...

115

AND DIGITAL IMAGES

SIGGRAPH

All information...

Information overload is already norm

"continuous partial attention"

People have easy access to more information than any world leader ever did until a few decades ago

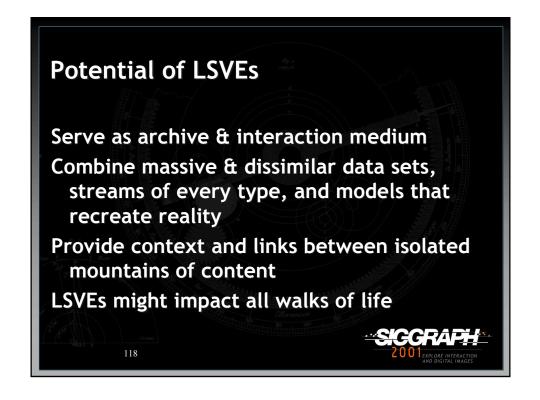
• (whether leaders use information is a separate question)

Scientific / information visualization plus hyperlinked 3-space can display, link all media

SICGRAPH 2001 EXPLOSE INTERACTION

116





Foundations for public cyberspace

Technical: VRML/X3D/Java3D, Java, IP
Political: Web3D Consortium stability
together with World Wide Web Consortium
(W3C) and Internet infrastructure
Economic: business cases already in play

Social: people issues are always hard to predict (and very fun to figure out)...

119



Personal impacts: coming soon

Imagine a world where the wonders of interactive 3D graphics are available to anyone anywhere anytime - not once a year to "chosen few" at SIGGRAPH...

Our community has long wondered about this possibility

Realization: networked 3D on the Web is now letting us find out

120



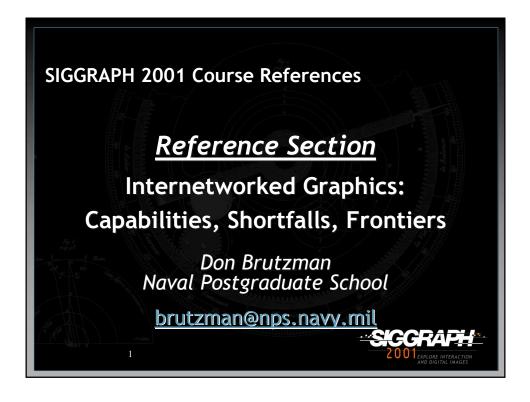


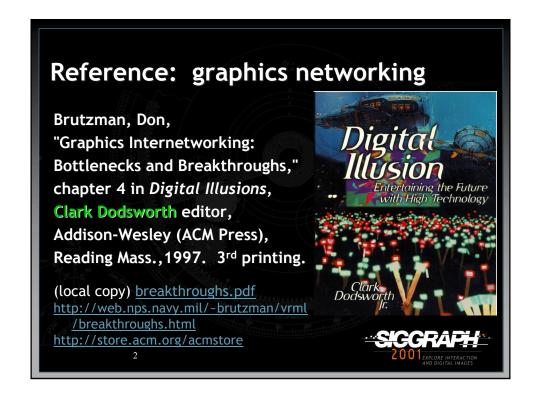


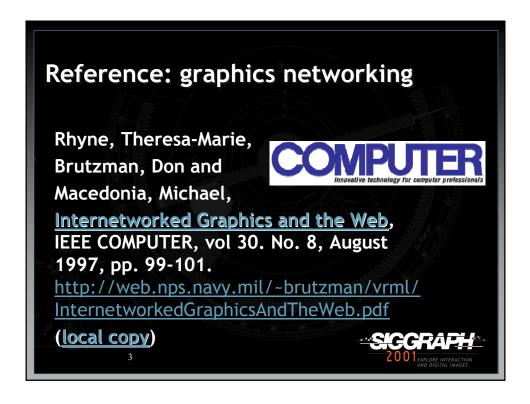


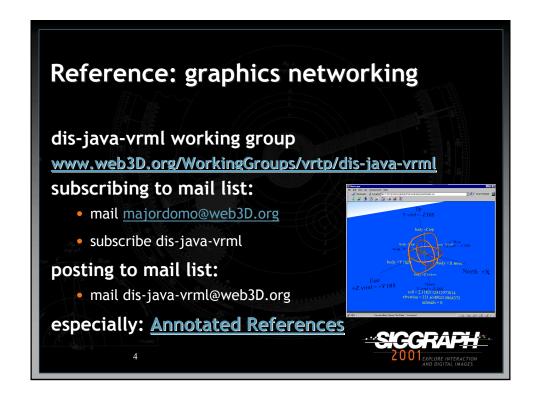












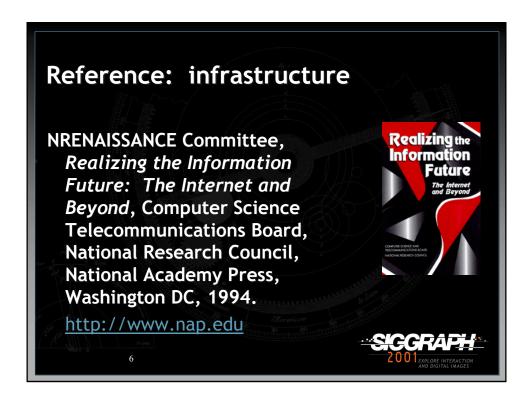
Key resource for developers: UCL book on multicast networks

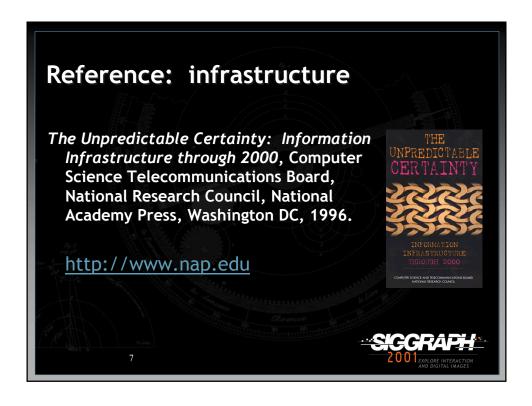
Internetworking Multimedia

- Jon Crowcroft, Mark Handley, Ian Wakeman
- UCL Press, December 1998
- http://www.cs.ucl.ac.uk/staff/jon/mmbook/book/book/book.html
- Hardcopy, Taylor & Francis Inc. 1999
 http://www.taylorandfrancis.com

5

SEGRAPIA 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

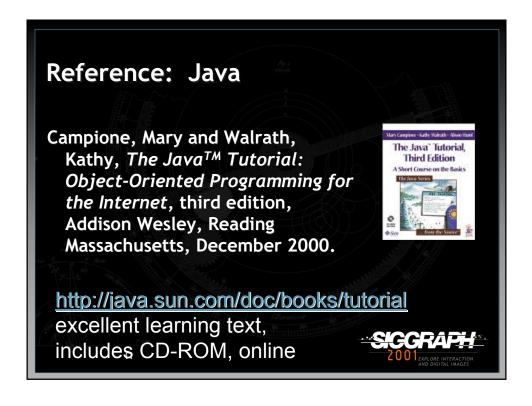














Reference: Java 3D

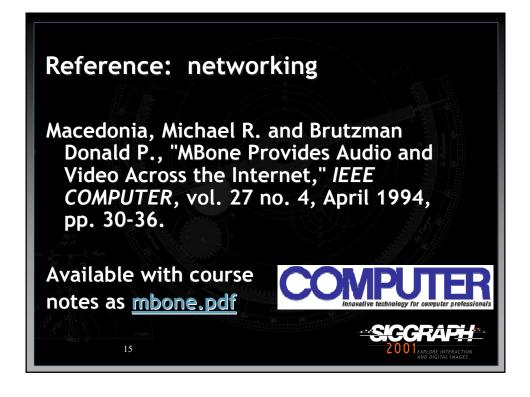
The Java 3DTM API is a set of classes for writing three dimensional graphics applications and 3D applets. It gives developers high level constructs for creating and manipulating 3D geometry and for constructing the structures used in rendering that geometry. Application developers can describe very large virtual worlds using these constructs, which provides Java 3D with enough information to render these worlds efficiently.

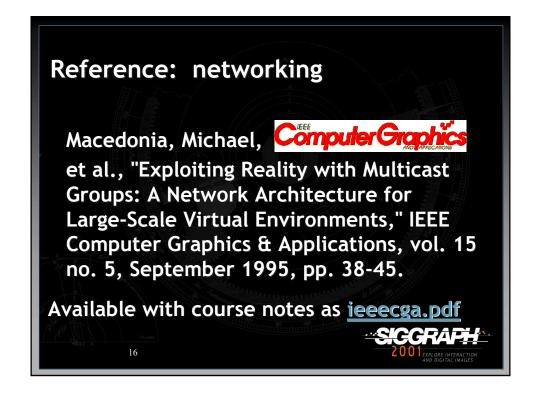
http://java.sun.com/products/java-media/3D

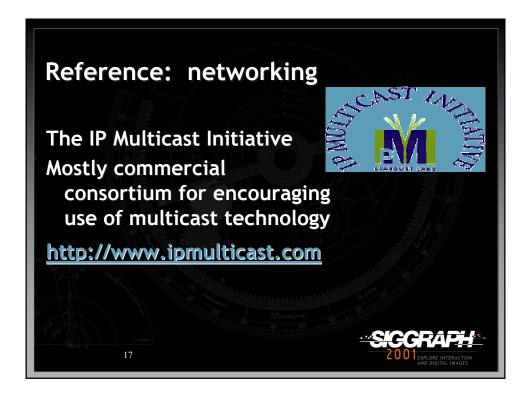


13

Reference: modeling/simulation Zyda, Michael and Sheehan, Jerry, editors, Modeling and Simulation: Competitiveness through Collaboration, Computer Science and Telecommunications Board, National Research Council, 1997. http://www.nap.edu SECCRAFIL 2001 RESOURCE 2001 RESOURCE

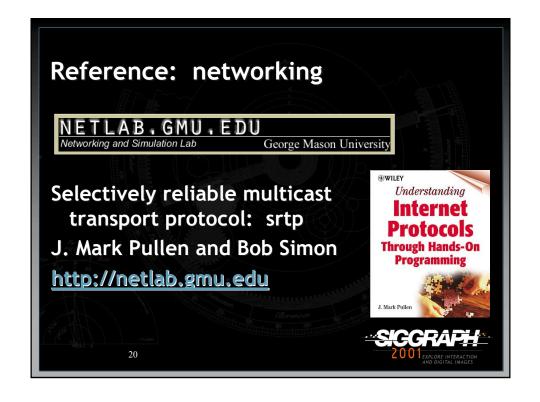


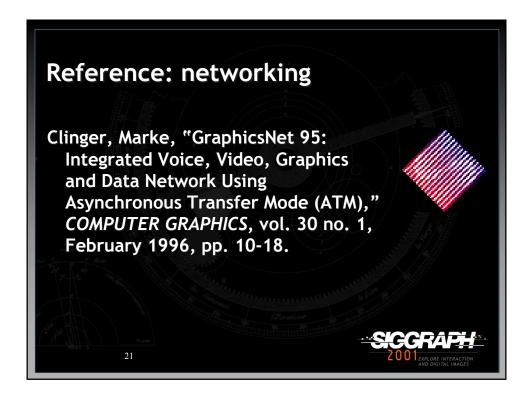


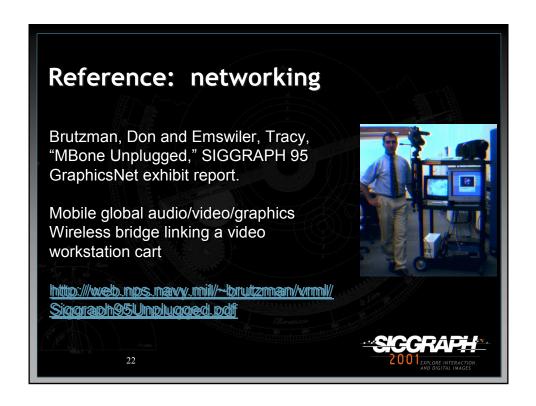












Reference: networking

Hoffman, Eric, Networking Services for Shared Virtual Worlds, tutorial notes, VRML 97: Second Symposium on the Virtual Reality Modeling Language, Monterey California, February 14 1997.

- provided on VRML 97 CD-ROM
- concise overview of concepts and issues, assumes networking knowledge, on CD-ROM

23

Reference: networking

Peterson, Larry L. and Davie, Bruce S., Networks: A Systems Approach, Morgan Kaufmann Publishers Inc., San Francisco California, Second Edition, October 1999.



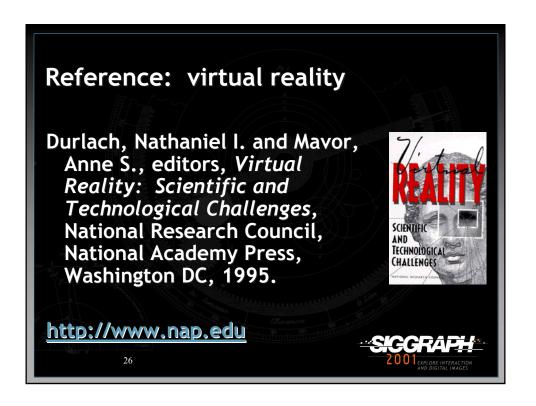
-SICGRAP

http://www.mkp.com - includes C software at http://www.cs.arizona.edu/xkernel

Networking book written from the perspective of Internet Protocol (IP) & working code

24



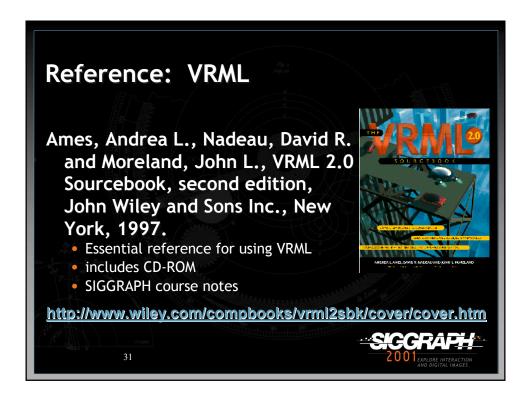


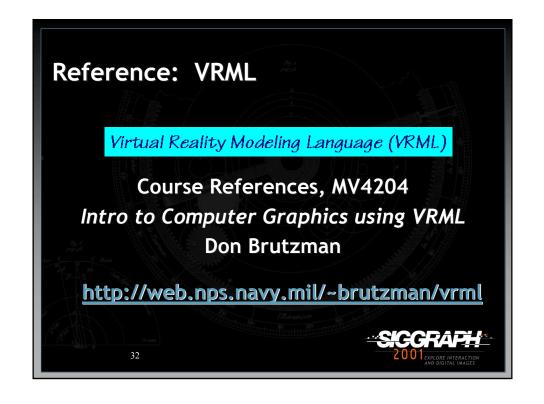




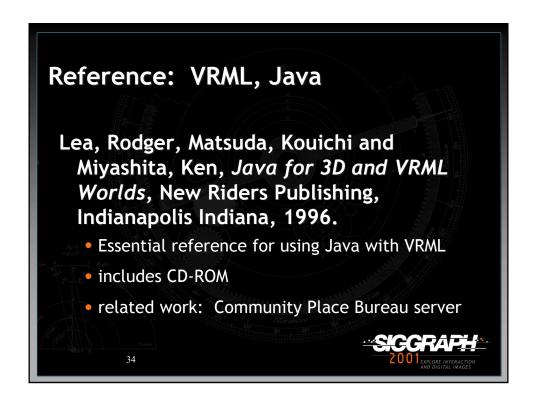




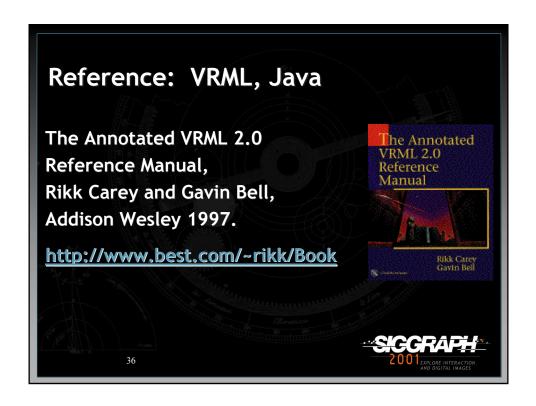








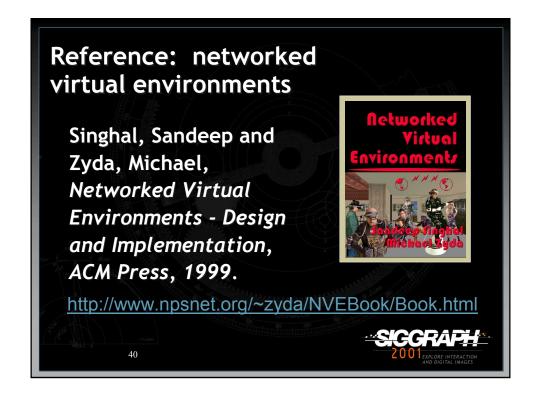






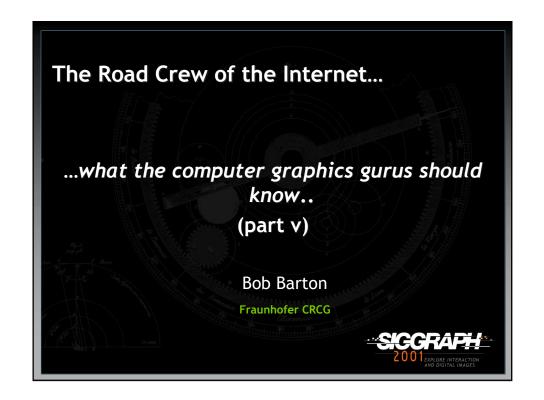








SIGGRAPH '01 Course Notes Integrating Standards with 3D Programming Paradigms & Issues in Distributed Virtual Environments Robert J Barton III Fraunhofer CRCG 01.401.453.6363 rbarton@crcg.edu Dr.-Ing. Norbert Schiffner Fraunhofer CRCG 01.401.453.6363 schiffner@crcg.edu



We will explore....

An update on MPEG-4 tools and support Status of MPEG-7

Distributed Team Training (purely web..)

Implementing modern protocols in CVEs - *Mediazine*

Collaborative AudioGraphics - Virtual Music Reproduction



For an Orientation and Overview of the IETF see:

www.crcg.edu/events/sig97/course.html

The Internet Engineering Task Force (IETF)

Organization and Standards

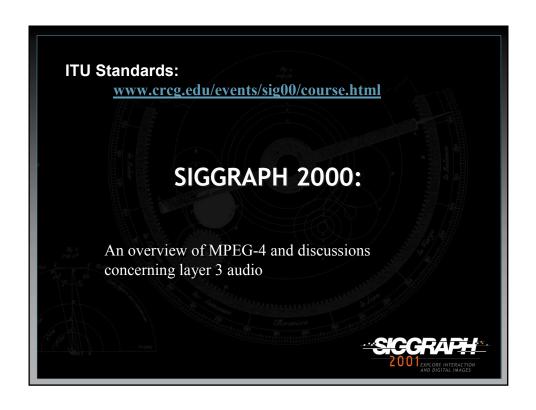
Bob Barton

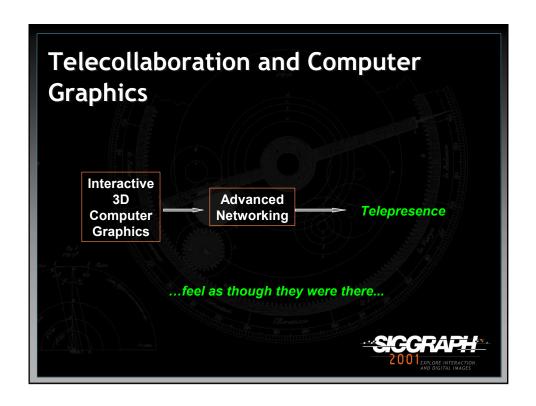
Fraunhofer CRCG

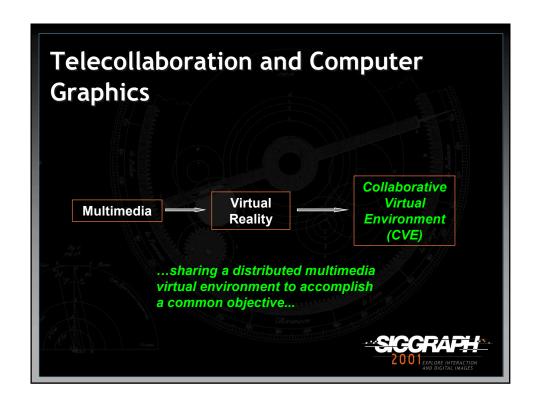


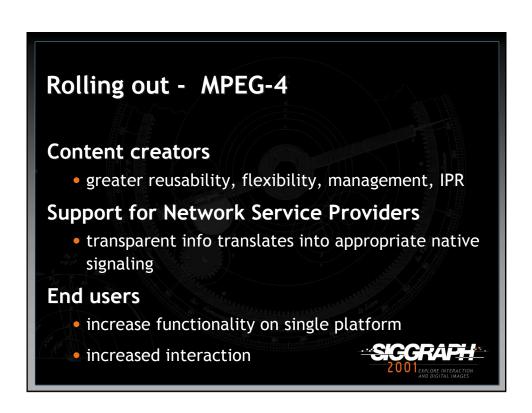




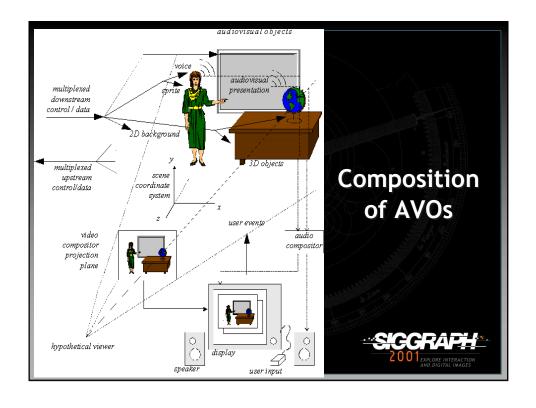








MPEG-4 Basics... Audio Visual Objects - primitive AVOs • natural or synthetic (recorded or generated) Compound primitives into AV scenes Multiplexing and synch. of data associated to AVOs Interactivity with the AV scene by the user



The Value of Information

AV info in digital archives, personal/professional databases, broadcast data streams, www...

Value depends on

- easy to find
- accessibility
- retrievability
- manageability

Access is increasing, discovery and management is becoming more difficult



MPEG-7 (Multimedia Content Description Interface)

Provide a rich set of tools to describe multimedia content

Targeted at both human users and automatic systems

Active participants:

• broadcasters, elex manufacturers, content creators, IPR managers, telecoms, academia



MPEG-7 Objectives

Extend proprietary solutions that exist today ie, data types

A standard set of descriptors

Defining custom structures - description schemes

A description definition language (DDL) for specifying new schemes

In a way, build upon MPEG-4



Levels of Abstraction in Descriptors

interaction

Descriptive features are user domain and app dependent - meaningful context abstraction level is related to feature extraction - automated vs human

An example: a barking dog and a moving ball



Other good info...

The form - raw data encoding type, size, etc.

Access conditions & rights

Classification - pre-defined categories

Links to other releavent material

Context - eg, non-fiction - Olympics 1996, decathlon, etc.



MPEG-7 Scope

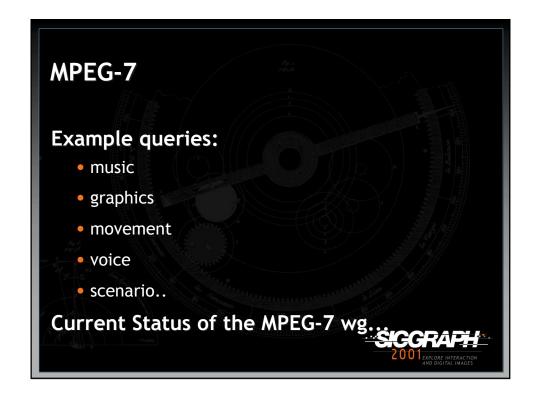
Standardizing representation that can be used for description - Meta Data stuff...

• smells like....

Does not standardize tools to generate descriptions, nor specify search engines or search techniques

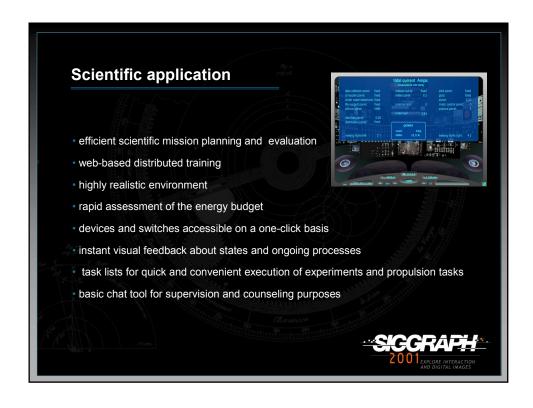


Applications like: • digital libraries, MM directory services, broadcast media channel selection, MM editing Domains like: • education, journalism, tourism, cultural heritage, entertainment, investigation services, GIS, remote sensing and surveillance, bio-med, shopping, film & video archives, ...



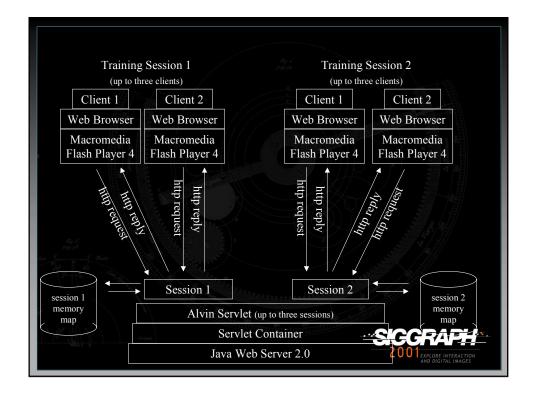








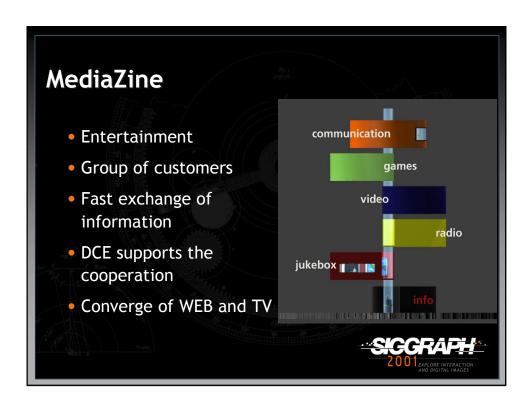






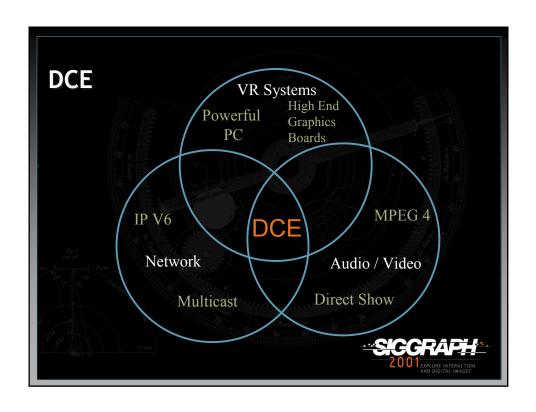




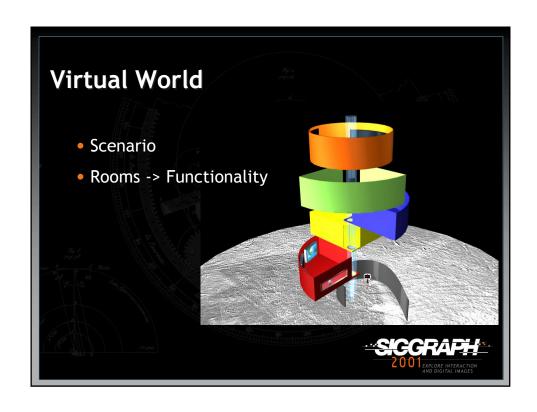




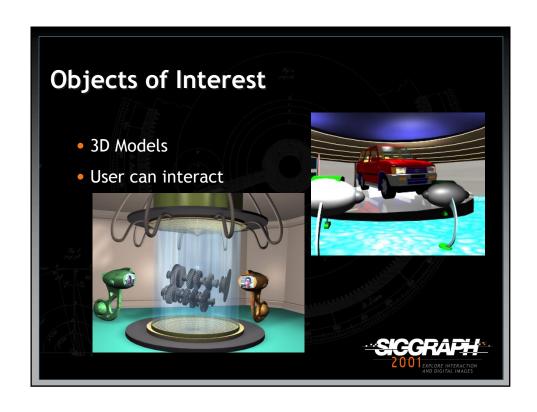


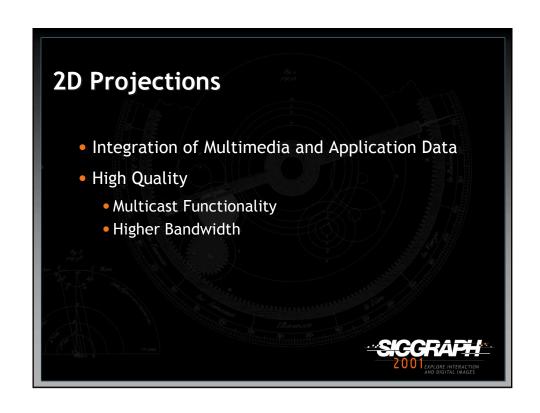




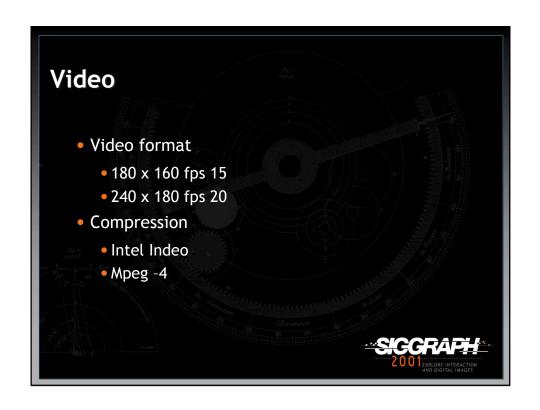


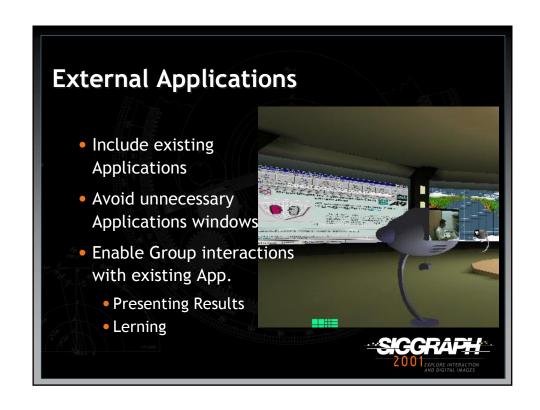


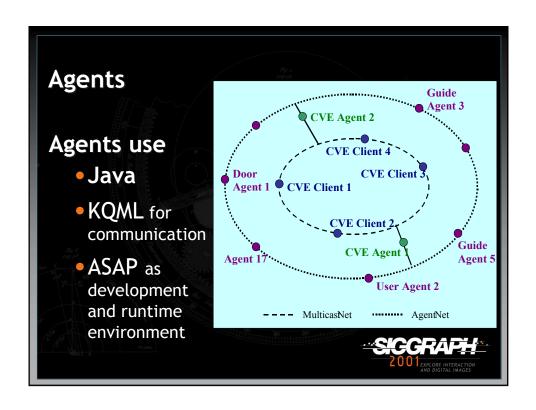




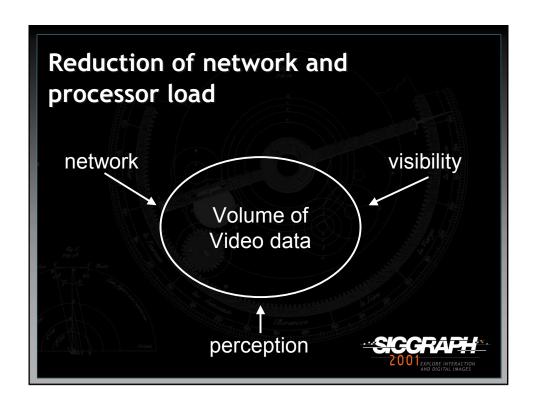
Spatial Audio Good Quality for communication Skhz (12/16/22) Robust against losses Supports active and passive sound sources Active sound is used for avatars Passive sound is used for environmental sound sources Control C

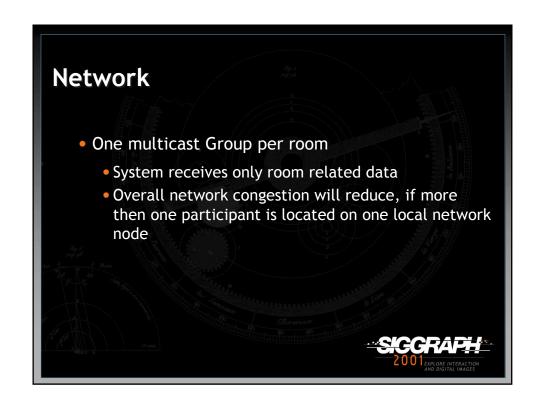








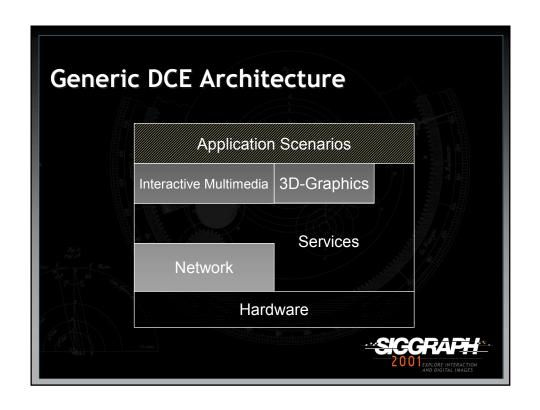


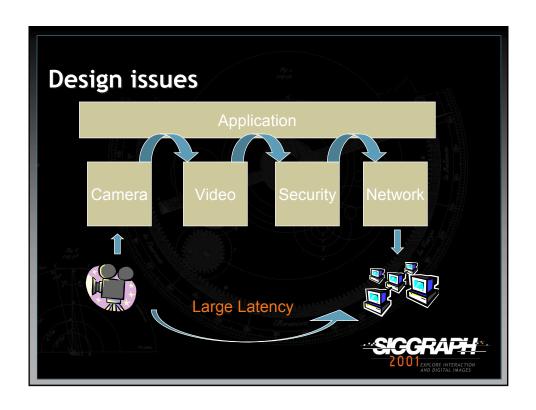


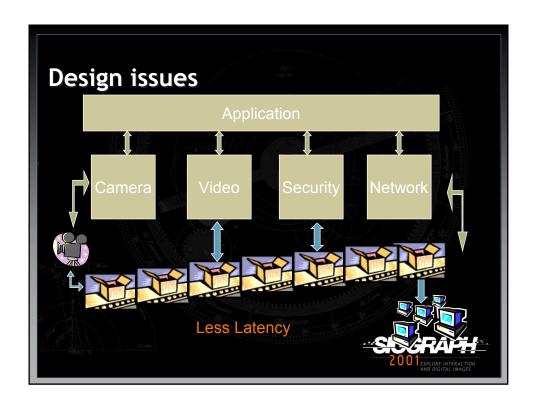


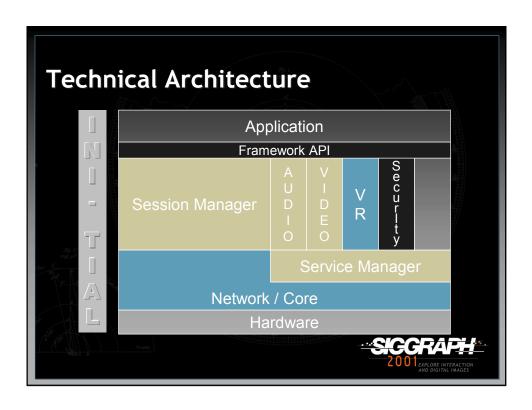
Perception • Video Quality depends • Distant between avatar and Video wall • Actions • Reduction of Frame per Second • Reduction of Pixel (Width * Height) • Changing from Live Video to freeze image













Distributed Experiences

Michael R. Macedonia michael_macedonia@stricom.army.mil

Why

- · Because it can be done
 - 3D graphics
 - Streaming media (voice, video, music)
 - Internet
 - Processors
 - Devices
- Market
 - Intel, Microsoft, Sony, AOL/EA
 - Games

Applications

- Military Simulation
 - CCTT, DIS, SIMNET
 - High Level Architecture
 - http://www.stricom.army.mil/

Applications

- Games
 - Java
 - Multiplayer FPS
 - ◆ Quake
 - Classical
 - RPGs
 - ◆ Everquest

Online Game Web Sites

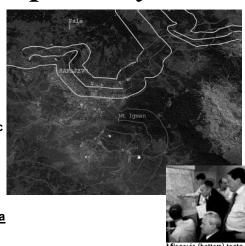
- · Zone.com
- · ultimaonline.com
- · mplayer.com
- · pogo.com
- heat.net

Applications

- Web Simulation
 - http://www.cise.ufl.edu/~fishwick/websim.html
- · Distributed Learning
- · Concurrent Engineering and CAD
- · Virtual Conferencing
- · Virtual Diplomacy
- · Data Mining
- · Telemedicine

Virtual Diplomacy

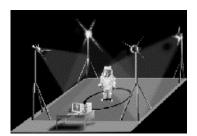
- November 28, 1995 from Correspondent Anthony Collings
- WASHINGTON (CNN) -Picture this -- Serbian
 President Slobodan Milosevic
 using a joystick to navigate a
 virtual reality map of Bosnia
 during the peace talks in
 Dayton, Ohio.
- http://wwwcgi.cnn.com/TECH/9511/virtua I bosnia/



Milosevic (bottom) tests virtual mapping device

Powerscene from Cambridge Associates

Virtual Emergency Response Training System (VERTS)







VERTS

Network Role Playing Games





Everquest

Asheron's Call

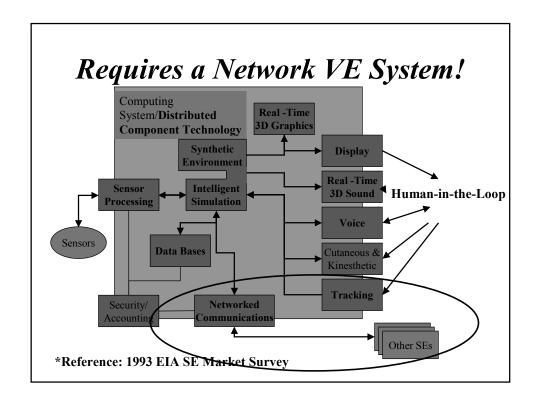
CAD







PTC Division Reality



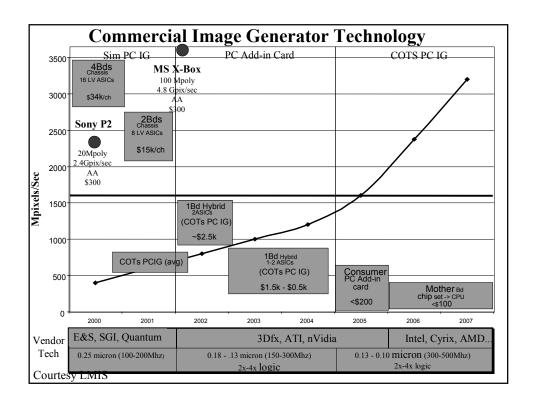
System Components Rapidly Advancing

- · Graphics
- · Audio
- Display
- Network

Courtesy Mitre

An Era of \$300 Supercomputers





The Playstation 3

- GScube
- 16 Playstation 2
 Boards in a <u>12 inch</u>
 Cube
- Latest SGI Origin 3400 File Server
- 65 Gflops (96 Gflops possible)





HMD Technology - Prognosis

- Market drivers: portable projectors, portable communications devices, wearable computers (medical, soldiers, road warriors, maintenance), consumer (view finders, games)
- Availability of the ideal HMD product: several years away
- · But lots of energy going into micro displays ...
- How do I do VR on a cell phone?

Head Mounted Displays





Nvision

Kaiser Electro Optics

Olympus EyeTrek



<u>FMD-250</u>

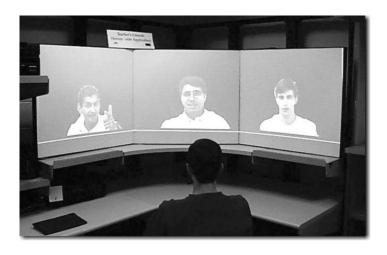
NTT DoCoMo I-mode Phone



Spatial Sound - Technical Challenges

- Developing accurate and adjustable Head-Related Transfer Functions (HRTF)
- Earphone design comfortable, nonintrusive and perfect match to ear
- Digital signal processing (real-time performance)
- · Incorporate echo and reverberation effects
- Extending to team training (multiple participants) (e.g. USC and TMH)

Immersistation



IMSC and NXT

Network Issues

- Bandwidth
- Network Distribution
- Latency
- Reliability

Bandwidth

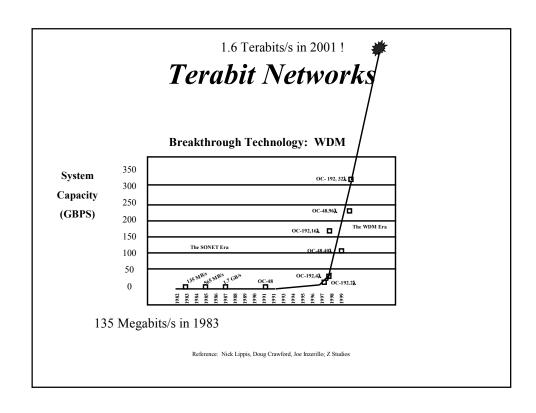
- Influenced by what you need and what you got:
 - LAN/WAN to Local Loop
 - ◆ Microsoft Zone Vs. Internet 2
 - Type of communication
 - ◆ e.g. stereo video Vs. entity state changes
 - ◆ DOOM Vs. DIS
 - ◆ ASCII Vs. compressed binary polygons
 - Number of participants/nodes
 - Distribution (multicast Vs. mesh network)
 - Dead-reckoning techniques

What's Happening

- · The local loop got faster
 - DSL
 - Cable modems
- Home networks
 - http://www.webproforum.com/home net/
- PS2 and X-Box designed for broadband
- Firewire (1394) and USB (400 Mb/s)
- LANs will get faster (Gigabit Ethernet)
- Wireless
 - Bluetooth
 - 3G Cell phones
 - 802.11

What's Happening

- · WANS will get very fast
- But routing bandwidth and DNS are problematic
- · In five years:
 - majority of US, Japan, and Europe will have broadband in the home
 - Three or four networked computers in the house means home LANS
 - wireless will become a major factor (e.g. Qualcomm, Nokia, NTT DoCoMo)



Sony's Pursuit of Broadband

- Sony, Tokyu tie up for advanced broadband services TOKYO, March 27 (Reuters) - Japan's Sony Corp, Tokyu Corp and unlisted Tokyu Cable Television Co said on Tuesday they will join hands to offer advanced broadband Net access from later this year.
- The company also said an agreement with wireless phone operator NTT DoCoMo in Japan will link PlayStation with DoCoMo's i-mode mobile service, and a broadbandcompatible phone will be introduced this summer.

AOL

Wednesday April 4, 4:33 pm Eastern Time

AOL Time Warner, Yahoo to benefit from broadband-report

NEW YORK, April 4 (Reuters) - AOL Time Warner Inc. (NYSE:AOL - news) and Yahoo Inc. (NasdaqNM:YHOO - news) are best positioned to benefit from the increasing penetration of high-speed access in the home, according to a report released by JP Morgan & McKinsey & Co.

Two-way high-speed Web access, especially through cable models, will also enable the integration of TVprogramming with interactive content and services. Jupiter Media Metrix expects interactive television services to penetrate about 42 percent of homes by 2005 compared to 8 percent by the end of this year.

"We expect AOL Time Warner with its (AOLTV service) and Microsoft (with its Ultimate TV service) to be among the most aggressive in this area," the report said.

Microsoft

Thursday March 29 4:54 PM ET

Microsoft, NTT Align For Xbox

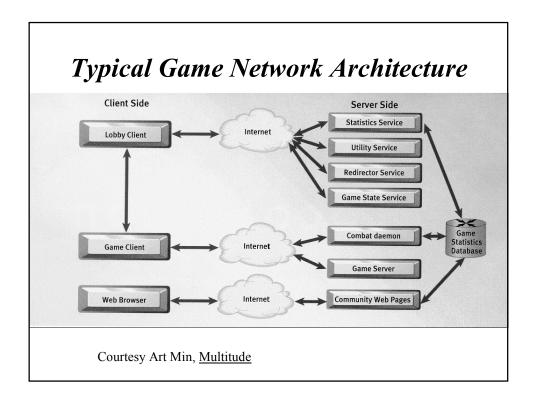
By MAY WONG, AP Technology Writer

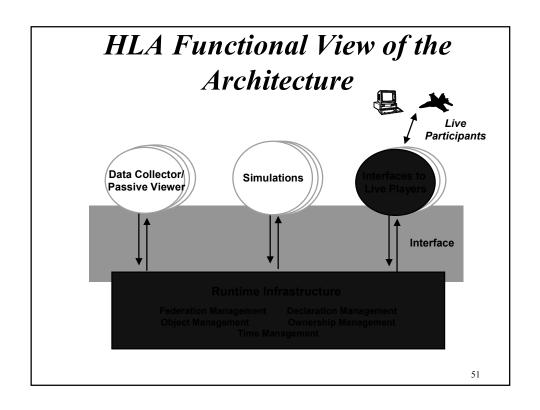
In a highly competitive move on its chief rival's home turf, Microsoft Corp. announced a deal Thursday with NTT Communications Corp. that will give users of Microsoft's Xbox (news - web sites) an online gaming service over a Japanese broadband network.



Network Distribution

- · Point-to-point -- Shared CAD viewer
- · Broadcast -- DIS
- · Multicast --
 - Networks: MBONE, UUNET
 - Applications: Video Conferencing, RealAudio
 - Client-Server
 - Java, Web, VRML, RPG, Half-Life
 - Capacity per server: 250-2500 players depending
- Mixed
 - T.120





Trends

- · Almost every game site is built on client server
- Huge server farms
- · Private networks and portals, e.g. EA and AOL
- We will have to wait for broadband for multicast
- Scaling will force content providers to use multicast

Latency and Jitter

- · The dark side of networked VR
- · Inherent in the system
- · You are always out-of-synch!

Latency and Jitter

· Influenced by:

- Distance, pure link speed is 2/3 speed of light (60 ms to Europe, 140 ms from Europe to Australia),
 10 min to Mars
- Network protocols (TCP, UDP, PPP, ATM)
- Compression
- Routing network congestion (TCP --> Jitter = Variable Delay)
- Software in the middle! (The operating system)
- Input and output devices (serialization)
- Packet sequencing

Fixes

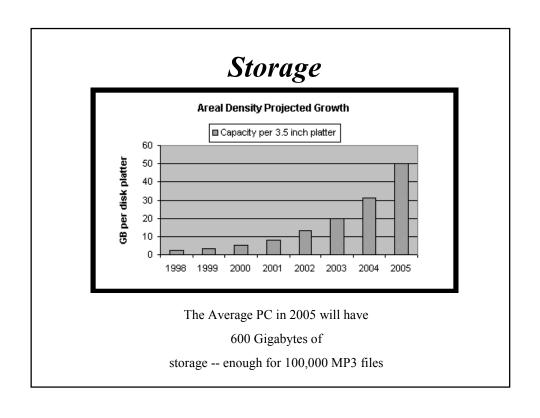
- Dead reckoning with smoothing (DIS)
- Measure latency with RTCP or PING
- · Synchronization and buffering with RTP
- Quality of service guarantees
- · Game Industry solutions
 - specialize the network (Valve/CISCO)
 - if its bad, cut you off
 - localized arenas
 - hope the Internet gets better (actually is)

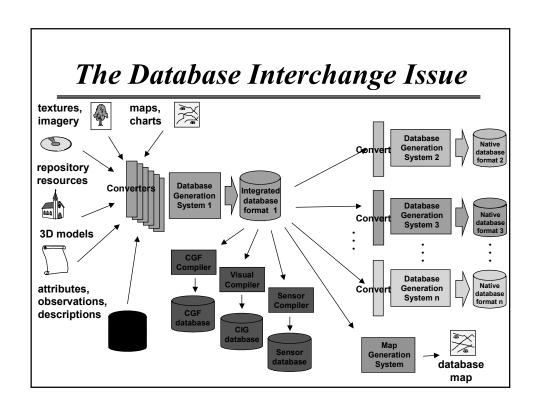
CISCO and Valve Powerplay

- Effort to improve multiplayer game support
- Intended to provide bandwidth reservation, low latency, and support for voice
- http://www.powerplayinfo.com/

Data Models for VEs

- Replicated homogeneous world database with peer updates
 - SIMNET, DIS, NPSNET
- · Shared, centralized server
 - MUDs. Quake
- · Shared, distributed databases with peer updates
 - Linda, ISIS
- Distributed, client-server databases with a local cache
 - VRML (WEB)





Opinion

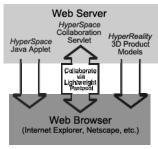
- · Really need a mix
 - Some homogenous data on client to get you started
 - Peer updates for lightweight data (entity state and audio)
 - Client-Server for initial data sets, models, behaviors
- · Database interchange
 - SEDRIS, Java3D, OpenGIS
- Data partioning scheme
 - -Area of Interest Manager

Processes

- Homogeneous
 - Most distibuted VE's
- Heterogeneous
 - DIS/HLA -- enforced by application protocol
- Migratory
 - JAVA, Javascript, Active X
 - Why security will become a bigger issue

Kaon

- · 3D graphics without plug-in
- · E-Commerce
 - http://www.kaon.com/
- · 30K Java-applet



Javagaming.org

- · Sun-sponsored
- Other companies: BLAM!, EA.Com, FriendlyGiants, Full Sail Real World Education, Games.Com, GameSpy Industries, Liquid Edge Games, Plazmic and Sony On-Line Entertainment
- · Rationale:
 - Web based gaming ability to instantly access high quality, communication based games.
 - Device independence access to game play from a wide variety of network enabled devices including TV set-top boxes, game consoles, PCs and cell phones.
 - Personalized content ability to easily modify gameplay elements such as story line, game levels and characters combined with chat or instant messaging to increase user experience.

Network Challenges

- strong data consistency (hard)
- · causality (really hard)
- · reliable (hard)
- · scalability
- · real-time (hard)
- · multicast
- · low-bandwidth
- · audio and video
- security (harder)

Other Trends

- A world of distributed supercomputing from the home
- Napster, Gnutella, and Freenet and anarchy
- Cultural barriers are disappearing among the young (e.g. success of AOL IM)

Summary

Building networked VEs is hard but powerful hardware and broadband are leading the way for ubiquitous VR.

- Amselem, Denis., "A Window on Shared Virtual Environments," Presence, 4, 2, Spring 95.
- Batsell, Stephen G. Batsell, "The Bi-level Multicast Architecture," DIS Communications Architecture Subgroup Winter Workshop, 18 January 1995.
- Benford, S., Bowers, J., Fahlen, L. and Greenhalgh, C., "Managing Mutual Awareness in Collaborative Virtual Environments", In Proceedings of VRST94, World Scientific Publishing Company, NJ, pp. 223-236.
- Borenstein, Nathaniel and Rose, Marshall T.," MIME Extensions for Mail-Enabled Applications," Internet working draft, ftp://thumper.bellcore.com/pub/nsb/st/safe-tcl.txt, October 1993s.
- Bouwens, Chris, "CAS and CCAS Meeting Notes from the 9th Workshop on Standards for the Interoperability of Defense Simulations", 13-17 September 1993, pp. 295-312.
- Bricken, William and Coco, Geoffrey, "The Veos Project", ftp.u.washington.edu:/public/VirtualReality/HITL/papers/techreports/Veos Project.ps, 1993.
- Burka, Lauren, "The MUD Archive, "http://www.ccs.neu.edu/home/lpb/muddex.html.

- Brutzman, Donald P., Macedonia, Michael R., and Zyda, Michael J.,

 "Internetwork Infrastructure Requirements for Virtual Environments,"
 submitted to the NII 2000 Forum of the Computer Science and
 Telecommunications Board, National Research Council, May 23-24, 1995,
 Washington, D.C.
- Brutzman, Donald P., Macedonia, Michael R., Zyda, Michael J.,
 Internetwork Infrastructure Requirements for Virtual Environments," in
 the *Proceedings of the Virtual Reality Modeling Language (VRML)*Symposium, San Diego Supercomputer Center (SDSC), San Diego, CA,
 December 13-15, 1995.
- Brutzman, Zyda, Watsen, Macedonia, virtual reality transfer protocol (vrtp)
 Design Rationale, in the Proceedings of the IEEE Sixth Workshop on
 Enabling Technologies, MIT, Cambridge, MA, June 18-20, 1997.
- Brutzman, D., Zyda, M., Macedonia, M., Luck, R., "A Virtual World for an Autonomous UnderWater Vehicle," in "The Edge," *Computer Graphics Visual Proceedings*, ACM SIGGRAPH '94, 1994.

- Carlsson, C. and Hagsand, O. 1993, "DIVE a Multi User Virtual Reality System," Proceedings of VRAIS93, September 18-22, Seattle, WA IEEE, NJ, 1993. pp. 394-400
- Cater, John P, "Use of the Remote Access Virtual Environment RAVEN for Coordinated IVA-EVA Astronaut Training and Evaluation," Presence, 4,1, Winter 1994
- Curtis, P., Nichols, D.A, "MUDs Grow Up: Social Virtual Reality in the Real World," 1994, ftp://ftp.parc.xerox.com/pub/MOO/papers/MUDsGrowUp.ps.
- Danset, Paul, "The GreenSpace Project," http://www.hitl.washington.edu/projects/greenspace.
- Durlach, Nathanliel I. and Mavor, Anne S., Virtual Reality: Scientific and Technological Challenges, National Academy Press, Washington, D.C. 1995.
- Gisi, Mark A., Sacchi, Cristiano, "Co-CAD: A Multi-user Collaborative Mechanical CAD System," Presence, 3, 4, Winter 1994.
- Goettsch, Ammo, Newton, Greg, Barton, Robert, and Macedonia, Michael R., A Real-Time Distributed Multimedia Virtual Environment for Collaboration, *Proceedings* of the First Int. Immersive Projection Technology Workshop, IAO, Stuttgart, Germany, 14-15 July 1997.

- Deering, Stephen, "MBONE-The Multicast Backbone," CERFnet Seminar, 3 March 1993.
- Gossweiler, Rich, Laferriere Robert J., Keller, Michael L. and Pauch, Randy, "An Introductory Tutorial for Developing Multiuser Virtual Environments," Presence, 3, 4 Winter 1994.
- Greenhalgh, Chris, and Benford, Steve, "MASSIVE: a Collaborative Virtual Environment for Tele-Conferencing," submitted to ACM TOCHI, 1994.
- Habib, Ibrahim W. and Saadawi, Tarek N., "Controlling Flow and Avoiding Congestion in Broadband Networks," IEEE Communications Magazine, Vol. 29, No. 10, October 1991, pp. 46-53.
- Kalawsky, Roy, The Science of Virtual Reality and Virtual Environments, Addison-Wesley, Workingham, England, 1993.
- Macedonia, Michael R., "From Personal to Transparent Computing", Fraunhofer International Symposium, "Computer Graphics in the Next 50 Years of Computing", Darmstadt, Germany, 30 Oct. 97.

- Noll, Stefan, Macedonia, Michael R., Chodura, Hartmut, Gärtner, Mathias, Schiffner, Norbert: Global ATM Testbed and Applications, Proceedings of TELECOM Inter@ctive 97, International Telecommunication Union, Place de Nations, CH-1211 Geneva 20, Switzerland, Geneva, 8-14 September 1997
- Macedonia, Michael R. and Zyda, Michael J., A Taxonomy for Networked Virtual Environments, *IEEE Multimedia*, March 1997, pp. 48-56.
- Macedonia, Michael and Noll, Stefan, A Transatlantic Research and Development Environment for Distributed 3D Graphics Application, *IEEE Computer Graphics and Applications*, March 1997, pp. 76-82.
- Macedonia, Michael R., Loftin, Bowen, Ziegler, Rolf, "Where No One Has Gone Before," in "The Digital Bayou," *Computer Graphics Visual Proceedings*, ACM SIGGRAPH '96, p.96, August 1996.
- Macedonia, Michael R. "Infrastructure for Global Virtual Environments", Panel on Global Multiuser Virtual Environments, *Proceedings of ACM SIGGRAPH '96*, pp. 495-496, August 1996.

Macedonia, Michael R., Zyda, Michael J., Pratt, David R., Brutzman, Donald P., Barham, Paul T., "Exploiting Reality with Multicast Groups: A Network Architecture for Large-Sca le Virtual Environments," in IEEE Computer Graphics and Applications, September 1995, pp. 38-45.

Macedonia, Michael R., "http://www

npsnet.cs.nps.navy.mil/npsnet/publications/Michael.Macedonia.thesis.ps.Z."A Network Software Architecture for Large Scale Virtual Environments," Ph.D. Dissertation, Naval Postgraduate School, Monterey, California, June, 1995.

Macedonia, Michael R., Brutzman, Donald P., Zyda, Michael J., Pratt, David R., Barham, Paul T., Falby, John and Locke, John, "NPSNET:A Multi-Player 3D Virtual Environment Over the Internet," in the *Proceedings of the 1995 ACM Symposium on Interactive* 3D Graphics, April 9-12, 1995, Monterey, California.

References

- Macedonia, Michael R., Zyda, Michael J., Pratt, David R., Barham, Paul T. and Zeswitz, Steven. "NPSNET: A Network Software Architecture for Large Scale Virtual Environments," *Presence*, Vol. 3, No. 4.
- Macedonia, Michael R., Brutzman, Donald P., <u>MBone Provides Audio and Video Across the Internet</u>, *IEEE Computer*, April, 1994, pp. 30-36.
- Macedonia, Michael R., Zyda, Michael J., Pratt, David R., Brutzman, Donald P. and Barham, Paul T. "Building Large-Scale Virtual Environments: A Network Software Architecture," Proceedings of Industrial Virtual Reality '95, Tokyo, Japan, 28 30 June 1995, pp.18-28.
- McCabe, James D., "Data Communications Required for CAS Applications," Presence, 4, 2 Summer 1995.
- Milgram, David L., Managed-Channel Architecture, LMSC X9400082P, Lockheed Missiles and Space Company, Palo Alto, CA May 1994.
- Miller, Duncan C., Pope, Arthur C. and Waters, Roland M, "Long-Haul Networking of Simulators," Proceedings: Tenth Interservice/Industry Training Systems Conference, Orlando, Florida, December 1989, p. 2.

Netrek, http://www.cs.cmu.edu:8001/afs/cs/user/jch/netrek/udp.

- Ohya, Jun, Kitamura, Yasuichi, Takemura, Haruo, et. al., "Real-time Reproduction of 3D Human Images in Virtual Space Teleconferencing," Proceedings of IEEE Virtual Reality International Symposium, September 1993, pp. 408-414.
- Partridge, Craig, Gigabit Networking, Addison-Wesley, Reading, Massachusetts, 1994.
- Perlman, Radia, Interconnections: Bridges and Routers, Addison-Wesley, New York, 1992, p. 258.
- Pesce, M., "The Virtual Reality Modeling Language," http://www.eit.com/vrml/vrmlspec.html.
- Pope, Arthur, "The SIMNET Network and Protocols," BBN Report No. 7102, BBN Systems and Technologies, Cambridge, Massachusetts, July, 1989.
- Pullen, Mark, "Toward a Requirement Specification for A Selectively Reliable Transport Protocol," DIS Communications Architecture Subgroup Winter Workshop, 18 January 1995.
- Pullen, Mark, "Dual-Mode Multicast," DIS Communications Architecture Subgroup Winter Workshop, 18 January 1995.

- Reinhardt, Andy, "The Network With Smarts," BYTE, October 1994.
- Rich, C. et al., "Demonstration of an Interactive Multimedia Environment," IEEE Computer, Vol. 27, No. 12, Dec. 1994, pp. 15-22
- Roy, Tina, M., Cruz-Niera, Carolina, DeFanti, Thoma A., Sandin, Daniel J., "Steering a High Performance Computing Application from a Virtual Environment," Presence, 4,2 Summer 1995.
- Rudin, Harry and Williamson, Robin, "Faster, More Efficient Streamlined Protocols," IEEE Communications Magazine, Vol. 27, No. 6, June 1989, pp. 10-12.
- Ryne, Theresa-Marie, Brutzman, Donald, Macedonia, Michael, Internetworked Graphics and the Web, *IEEE Computer*, August 1997, pp. 99-101.
- Ryne, Theresa-Marie, Brutzman, Donald, Macedonia, Michael, Barton, Robert, "Beyond Bottlenecks and Roadblocks: Internetworked Computer Graphics", course for SIGGRAPH'97, August 1997.
- Sawler, Robert, Matusof, 'Issues Concerning Cue Correlation and Synchronous Networked Simulators,' AIAA, 1991.

- Sayers, Craig, and Paul, Richard, 'An Operator Interface for Teleprogramming Employing Synthetic Fixtures,' Presence, 3, 4, Winter 1994, pp. 309-320.
- Shaw, Chris, and Green Mark, 'The MR Toolkit Peers Package and Experiment,'
 Proceedings of IEEE Virtual Reality International Symposium, September 1993,
 pp. 463-469.
- Schroeder, Michael D. and Burrows, Michael, "Performance of Firefly RPC," SRC Research Report 43, 15 April 1989.
- Schilit, Bill N. and Theimer, Marvin M, "Disseminating Active Map Information to Mobile Hosts," IEEE Network, September 1994, pp. 22-32.
- Singh, Gurminder, "A Software Toolkit for Network-Based Virtual Environments," Presence, 3, 1, Summer1994.
- Singhal, Sandeep K., "Using a Position History-Based Protocol for Distributed Object Visualization.," in Designing Real-Time Graphics for Entertainment [Course Notes for SIGGRAPH '94 Course #14] July 1994.
- Stanford Research Institute International, ATD-1 Architecture White Paper Edit Draft, Menlo Park, CA, undated.
- Stanford Research Institute MAGIC Home Page, http://www.ai.sri.com:80/~magic/.

- Taubes, Gary, "Surgery in Cyberspace," Discover, December 1994, pp. 85-94.
- Van Hook, Daniel J., Calvin, James O., Newton, Michael K., Fusco, David A, "An Approach to DIS Scalability," 11th DIS Workshop, September 1994, pp. 347-355.
- Van Hook, Daniel J., Calvin, James O, AGENTS: An Architectural Construct to Support Distributed Simulation, 11th DIS Workshop, September 1994, pp. 357-366.
- Van Hook, Daniel J., Calvin, James O, "Approaches to Relevance Filtering," 11th DIS Workshop, September 1994, pp. 367-369.
- Wloka, Mathias M., "Lag in Multiprocessor VR," Presence, 4, 1, Spring 1995.
- Zyda, Michael J., Pratt, David R., Falby, John S., Barham, Paul T. and Kelleher, Kristen M., "NPSNET and the Naval Postgraduate School Graphics and Video Laboratory," Presence, 2, 3, Spring 1994.
- Zyda, Michael J., Pratt, David R., John S. Falby, Chuck Lombardo, Kelleher, Kristen M., "The Software Required for the Computer Generation of Virtual Environments," Presence, 2, 2, Spring 1993, pp. 130-140.

Backups

Classes of Data in the System

- Renderer
 - Synchronization between renderers
 - Lowest latency
- Agent
 - Communication between agents
 - Guaranteed delivery, flexibility
- Simulation
 - Distributed shared world state updates
 - Low latency, variable QOS

- · Device
 - Input/Output device events
 - Low latency, LAN use only
- · Live Media
 - Video textures, live audio
 - Like Internet conferencing

Reliability

- Need to guarantee some data
 will be received.
 Reliable (Gets there)
 - models.
 - behaviors,
 - controls,
 - management
- Real-time (On time)

Scaleable (Group size)

· Weak vs. strong-data consistency

Fixes

- · Some data is less critical.
 - Motion vectors
 - Voice packets
- Techniques
 - Replication (redundancy)
 - Variable guarantees
 - Network (QOS)

Views

- · How tightly coupled?
 - How to get two avatars to waltz
- Synchronous
 - CAVE
- Asynchronous
 - SIMNET/CCTT
- Mixed
 - Shared CAD

Opinion

- Highly coupled views over wide-area nets can be faked
 - Just don't be very interactive (play chess or role play)
 - Keep it first person (don't look at yourself)
 - No dancing

Graphics Internetworking: Bottlenecks and Breakthroughs

Don Brutzman
Code UW/Br, Naval Postgraduate School
Monterey California 93943 USA
831.656.2149 voice, 831.656.3679 fax
brutzman @nps.navy.mil

http://web.nps.navy.mil/~brutzman/vrml/breakthroughs.html

Chapter 4, *Digital Illusions*, <u>Clark Dodsworth</u> editor, Addison-Wesley, Reading Massachusetts, 1997, pp. 61-97.

1. Introduction and Motivation. Although networking is considered to be "different" than computer graphics, network considerations are integral to large-scale interactive three-dimensional (3D) graphics. Graphics and networks are now two interlocking halves of a greater whole: distributed virtual environments. New capabilities, new applications and new ideas abound in this rich intersection of critical technologies. Our ultimate goal is to use networked interactive 3D graphics to take full advantage of all computation, content and people resources available on the Internet.

Network breakthroughs repeatedly remove bottlenecks and provide new opportunities. A pattern appears as we attempt to scale up in capability and capacity without limit: every old bottleneck broken reveals another. Understanding the bottlenecks, the corresponding solutions and potential upper bounds to growth permits us to develop effective networked graphics. When we overcome current bottlenecks, "effectively networked graphics" will simply mean "applications."

Internetworked graphics can be examined from the perspectives of connectivity, content, interaction, economics, applications and personal impacts. Internetworking refers to the ability to seamlessly interconnect multiple dissimilar networks globally. Connectivity has numerous dimensions including capacity, bandwidth, protocols and the many-to-many capabilities of multicasting. Content equals the World Wide Web and includes any type of information, dataset or data stream that might be used in the graphics environment. Interaction implies minimal latency, a sense of presence, and the ability to both access and modify content. The economics of networked graphics environments is developing rapidly and principal forces can be identified. Applications drive infrastructure development and are the most exciting part of networked graphics. Finally, the personal impacts which accompany these developments range from trivial to profound as high-quality interactive internetworked computer graphics become the norm on all computers.

2. Connectivity. There are many component pieces to the network connectivity puzzle. We will examine physical connectivity, network layers, the Internet Protocol suite, multicast distribution, the Multicast Backbone (MBone), enabling individuals, and vertical integration. An extensive overview of many networking topics is necessary to provide fundamental technical background. An understanding of each of these components is needed if we are to successfully implement highly shared graphics applications.

Physical connectivity to the Internet is a prerequisite to internetworked computer graphics. Other types of connections do not scale. There are several types of Internet access, roughly corresponding to direct providers, online services and bulletin board systems (BBSs). Foremost is "direct" connectivity where a user's computer is connected via a telephone modem, local-area network (LAN) or other link to an Internet service provider. The native communication protocol used across these direct links and the global Internet is the Internet Protocol (IP). Designed for minimum complexity and maximum capability, IP compatibility and direct Internet connectivity are essential for any cutting-edge application that uses "anybody anywhere anytime any-kind" connectivity. Lesser degrees of connectivity are possible via gateways when indirectly connected through commercial services such as America On-Line (AOL) or CompuServe. Even lower degrees of connectivity like file transfer and store-and-forward e-mail are possible through dial-up BBSs. To achieve the maximum breakthroughs in networked computer graphics, direct Internet connectivity is essential. We assume such connectivity throughout this chapter.

Network protocols and layered models. To integrate networks with large-scale graphical virtual environments, we invoke underlying network functions from within applications. Figure 1 shows how the seven layers of the

1 of 23 4/16/99 11:07 AM

well-known Open Systems Interconnection (OSI) standard network model generally correspond to the effective layers of the Internet Protocol (IP) standard. Functional characteristic definitions of the IP layers follow in Figure 2. These diagrams and definitions are merely an overview to help illustrate the logical relationship and relative expense of different network interactions. OSI and IP are often viewed as competing approaches to networking. A good overview of relative strengths and weaknesses of OSI versus IP is presented in [Malamud 92]. In practice OSI-related protocols are not of interest due to typically greater overhead and closed standards. For graphics internetworking, IP compatibility is essential for reasons of performance, open specifications, development flexibility and overwhelmingly strong global deployment trends.

In general, network operations consume proportionately more processor cycles at the higher layers. We must minimize this computational burden to reduce latency and maintain real-time responsiveness. Two protocols are currently available in the transport layer: one ensures reliable delivery, the other makes no guarantees (therefore requiring less overhead). Information transfer must use either reliable connection-oriented Transport Control Protocol (TCP) or nonguaranteed-delivery connectionless User Datagram Protocol (UDP). We choose between the two protocols depending on the criticality, timeliness and cost of imposing reliable delivery on the particular stream being distributed. Understanding the precise characteristics of TCP, UDP and related protocols helps the virtual-world designer understand the strengths and weaknesses of each network tool employed. Since internetworking considerations impact all components in a large virtual environment, understanding network protocols and application hooks is essential for virtual-world designers [Internic 95] [Stallings 94] [Comer 91] [Stevens 90].

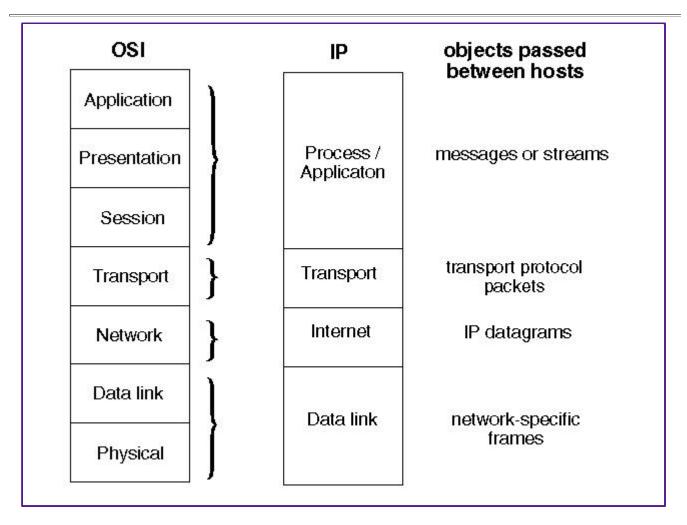


Figure 1. Correspondence between OSI and IP protocol layer models, and objects passed between corresponding host layers.

Internet Protocol (IP). Although the protocols associated with internetworking are diverse, there are some unifying concepts. Foremost is "IP on everything," the principle that every protocol coexists compatibly within the Internet

2 of 23 4/16/99 11:07 AM

Protocol suite. The global reach and collective momentum of IP-related protocols makes their use essential. Current IPv4 and next-generation IPv6 [Bradner 95] protocols can be run over a complete variety of electrical, radio-frequency and optical physical media.

Examination of protocol layers helps clarify network issues (Figure 2). The lowest layers (Physical, Data Link) are reasonably stable, with a huge installed base of Ethernet and FDDI systems, augmented by the rapid development of wireless and broadband ISDN solutions such as Asynchronous Transfer Mode (ATM). Compatibility with the Internet Protocol (IP) suite is assumed. The middle transport-related layers (Internet, Transport) are a busy research and development area. Real-time reliability, quality of service, and other capabilities are evolving rapidly due to competing market forces, both academic and commercial. Lower- and middle-layer bottlenecks are mostly solvable using available techniques. The major bottlenecks to large-scale graphics internetworking are at the application layer.

Process/Application Layer. Invokes TCP/IP services, sending and receiving messages or streams with other hosts. Delivery can be intermittent or continuous.

Transport Layer. Provides host-host packetized communication between applications, using either reliable-delivery connection-oriented TCP or unreliable-delivery connectionless UDP. Exchanges packets end-end with other hosts.

Internet/Network Layer. Encapsulates packets with an IP datagram that contains routing information, receives or ignores incoming datagrams as appropriate from other hosts. Checks datagram validity, handles network error and control messages.

Data Link/Physical Layer. Includes physical media signaling and lowest level hardware functions, exchanges network-specific data frames with other devices. Includes capability to screen multicast packets by port number at the hardware level.

Figure 2. Summary of Internet Protocol (IP) suite layer functionality.

Multicast. Multicast packet distribution is best understood in comparison with two other packet distribution mechanisms: unicast and broadcast. Unicast is point-to-point communication, commonly occurring whenever an e-mail message is sent or a Web browser connects to a home page. Only the recipient host and intermediate routers need to spend computational cycles on a unicast packet. One-to-many unicast communications are accomplished by multiple streams corresponding to every recipient. While this is not a problem when sending an e-mail message to multiple recipients, it is clearly an expensive proposition when sending a high-bandwidth stream such as graphics or video.

Broadcasting is at the opposite end of the spectrum from unicast. Broadcast messages reach every host on a local-area network and demand a response from each application or operating system. This is extremely inefficient at high bandwidths. Furthermore, each broadcast packet must be read and processed by every host on a network. Broadcast is typically prohibited from passing across routers to prevent such packets from propagating everywhere. Figure 3 shows typical unicast and broadcast packet stream paths.

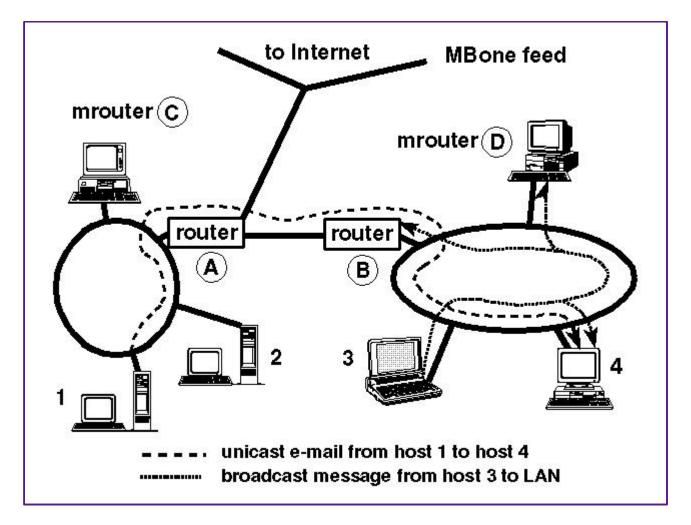


Figure 3. Typical unicast and broadcast packet stream paths.

Multicast packet functionality lies between unicast and broadcast. A single multicast packet can touch every host on a LAN without duplication. However each host can subscribe or ignore multicast at the hardware level by informing the network interface card which multicast channels (if any) to monitor. This feature is extremely important since a single high-bandwidth stream can still reach an arbitrary number of hosts, but computational load is only seen on hosts which explicitly subscribe to the multicast channel. Multicast packets are usually prevented from traversing routers to prevent overloading the Internet with arbitrary propagation of such streams. Figure 4 illustrates this behavior. To summarize, the key network characteristics which distinguish multicast from unicast and broadcast include:

- · individual multicast packets can be read by every host workstation on a LAN
- · elimination of duplicate streams minimizes bandwidth requirements
- workstations can screen unwanted multicast packets at the hardware level, eliminating unnecessary computational burden on applications

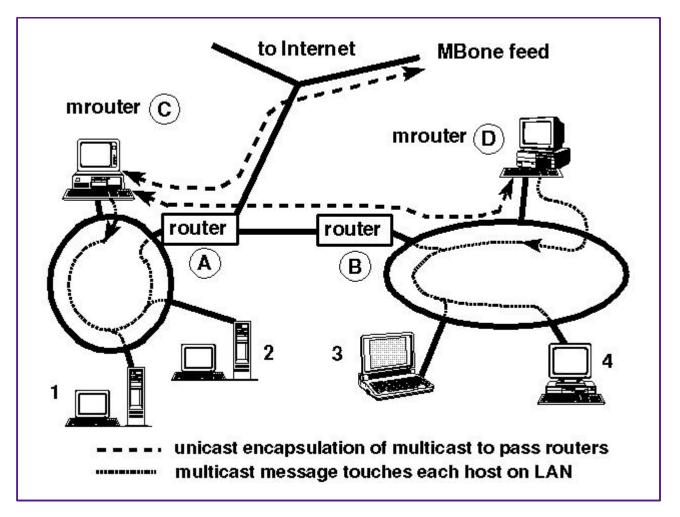


Figure 4. Example of multicast routers connecting multicast streams encapsulated by unicast (in order to bypass router prohibitions on multicast). After stripping unicast headers, multicast streams are then available to each host on the LAN. Direct multicast support in routers has recently become available. Such multicast routes between LANs comprise the Multicast Backbone (MBone).

Multicasting has existed for several years on local-area networks such as Ethernet and Fiber Distributed Data Interface (FDDI). Now, with Internet Protocol multicast addressing at the network layer, group communication can be established across the Internet. IP multicast addressing is an Internet standard developed by Steve Deering [Deering 89]. Categorized officially as an IP Class D address, an IP multicast address is merely a set of reserved IP addresses in the range (224.0.0.0 thru 239.255.255). Multicast is supported by numerous workstation vendors including SGI, Sun, DEC and HP.

Unicast audio/video applications connecting groups of people across the Internet can be harmful because they waste bandwidth and do not scale. Recent examples include PC and Macintosh implementations of CU-SeeMe [Cogger 95]. Multicast applications have long been expected for Intel and Apple processors, but first multitasking must be better implemented in personal computer operating systems. Current audio/video applications for personal computers are typically unicast and need to be modified to support multicast before widespread use of arbitrarily shared data streams can be expected for all computers on the Internet.

Multicast Backbone (MBone). The MBone is one of the Internet's most interesting capabilities and is used for distribution of live audio, video and other packets on a global scale. MBone is a virtual network because it shares the same physical media as the Internet. It uses a network of routers (mrouters) that can support multicast. These mrouters are either upgraded commercial routers, or dedicated workstations with modified operating system kernels running in parallel with standard routers. It was named by Steve Casner and originated from an effort in 1992 to multicast audio and video from meetings of the Internet Engineering Task Force (IETF). The fact that the MBone works at all is pretty remarkable since the original multicast backbone code was a student experiment. Encapsulated multicast streams sidestep regular routers through handcrafted tunnels across the Internet. Thus the

MBone exemplifies the best of the hacker ethic on a large scale. This pioneer spirit is still visible today as we cooperatively transition to a more carefully engineered backbone service that includes widespread multicast service.

Two things make multicasting feasible on a world-wide scale: installation of high-bandwidth Internet backbone connections, and widespread availability of workstations with adequate processing power and built-in audio capability. Today hundreds of researchers use MBone to develop protocols and applications for group communication. Multicast provides one-to-many, several-to-many and many-to-many network delivery services. A one-to-many example might be live transmission of an academic conference. A several-to-many event might be a panel interview between different locations attended by many dispersed people. A many-to-many example might be a distributed battlefield exercise with 500 active players. Thus multicast is useful for a variety of videoconferencing, audio and multiplayer events where numerous hosts need to communicate simultaneously. Network concepts underlying MBone, bandwidth considerations, application tools, MBone events, interesting MBone uses, and how to connect to the MBone are described in "MBone Provides Audio and Video across the Internet" [Macedonia, Brutzman 94].

Lost MBone packets stay lost. Since multicast currently uses only the User Datagram Protocol (UDP) and not the Transport Control Protocol (TCP) of the IP suite, multicast streams are connectionless "unreliable" service. No setup is required, no acknowledgements are used, and no guarantee of delivery exists for this type of "best-effort" service. This is ordinarily a good thing for most real-time information streams (such as audio and video) since it avoids delivery bottlenecks and unwanted overhead. Furthermore it doesn't matter what kind of data is sent via multicast IP packets. Several researchers are experimenting with "reliable multicast" transport protocols which try to achieve a different balance by gaining occasional retransmission benefits without unacceptable acknowledgement overheads [Schulzrinne 96]. These considerations are critical when scaling to arbitrarily large numbers of simultaneously interacting users on the network. A breakthrough in this bottleneck will enable new global communication paradigms for widely distributed shared applications.

The keys to understanding MBone constraints are capacity and bandwidth. The reason a multicast stream is bandwidth-efficient is that one packet can touch all workstations on a network. A 64-kilobit per second (Kbps) video stream typically provides 3 to 5 frames per second of quarter-size NTSC video. Each multicast stream uses the same bandwidth whether it is received by 1 workstation or 20. That is good. However, that same multicast packet is ordinarily prevented from crossing network boundaries such as routers. The reasons for this current restriction are religious and obvious from a networking standpoint. If a multicast stream that can touch every workstation jumps from network to network without controls, the entire Internet might quickly become saturated by such streams. That scenario is disastrous, so group controls are necessary. The MBone controls multicast packet distribution across the Internet in several ways. The topological "lifetime" of multicast packets is constrained to limit the number of allowed mrouter hops, and automatic pruning/grafting algorithms adaptively restrict multicast transmission to active participants. Such controls can prevent most (but not all) global MBone problems.

Responsible daily use of the MBone network merely consists of making sure you don't overload your local or regional bandwidth capacity. Appropriate bandwidth values initially seem obscure but daily practice quickly makes such figures intuitive. MBone protocol developers are successfully experimenting with automatically pruning and grafting subtrees, but for the most part MBone uses thresholds to truncate broadcasts to the leaf routers. The truncation is done by the setting the "time-to-live" (ttl) field in a packet that is decremented by at least one each time the packet passes though an mrouter. A ttl value of one prevents a multicast packet from escaping a LAN. An initial ttl value of 16 typically limits multicast to a campus, as opposed to values of 127 or 191, which might send a multicast stream to every subnet on the MBone (currently over 20 countries). Sometimes we decrement ttl fields by intentionally large predetermined values to limit multicasts to sites and regions.

These issues may sound challenging but are relatively uncomplicated in practice. Some personal technical proficiency is important because use of the MBone can have a major impact on shared network performance. For example, default video and audio streams consume about 200 Kbps of bandwidth, over 10 percent of a 1.5-Mbps T1 line (a common site-to-site link on the Internet). Several simultaneous high-bandwidth sessions might easily saturate network links and routers. The number of active users on current MBone channels is usually small enough that consensus and planning are able to effectively share this limited global resource. These low numbers will change soon. Planning, consensus and careful utilization of physical and logical network resources will remain essential components of live large-scale global internetworking.

"It is not every day that someone says to you, 'Here is a multimedia television station that you can use to broadcast from your desktop to the world.' These are powerful concepts and powerful tools that extend our ability to communicate and collaborate tremendously. They have already changed the way

people work and interact on the net." [Macedonia, Brutzman 94]

Enabling individuals. More and more often, the cost of admission for participation in global internetworking is a willingness to learn and participate. The cost of equipment is relatively low and the price of bandwidth continues to become more affordable in most countries. As physical infrastructure becomes more prevalent, physical connectivity becomes easily available for many people. Public domain software for almost any purpose can be obtained freely over the Internet. Frequently asked question lists (FAQs), mailing lists and even online books provide amazing depth on almost every subject related to networking and graphics. Individuals are now able to do many things that previously required large groups and large budgets to accomplish. This trend is a tremendous equalizer which lets students, researchers and companies become effective players in a global arena.

Vertical integration. For many years there have been walls separating communities of users corresponding to the computer architecture or operating system used. Supercomputers, mainframes, workstations and personal computers were so dissimilar that commonality between them was practically nonexistent. Networked applications have changed this situation completely. Once-isolated islands are now complementary, interdependent and practically seamless.

Initial rudimentary examples of interconnectivity are e-mail, file transfer protocol (ftp) and remote terminal protocol (telnet). The Internet Protocol suite enables standardized hostnames, host numbers and Uniform Resource Locators (URLs), which in turn convert loosely connected hardware into a shared global address space. Sharable applications such as Mosaic, Netscape, distributed interactive simulation and the MBone tools show that equivalent functionality can be expected on any computer platform, from the supercomputers to PCs. The Hypertext Markup Language (HTML) and the Web now provide a universal interface to any computational process, a fact surprisingly missed by most interface designers. This new interoperability between all computers means that applications designers can think in terms of vertical integration. Thousands of simultaneous users with networked personal computers can now query, drive and evaluate computationally intensive graphics scenes rendered by powerful computers at arbitrarily distant locations. Such scenarios have been described theoretically for years. While theoretical discussions continue, the practical capability has arrived. Together, all the countless machines interconnected by the Internet are our new supercomputer.

Here are the success metrics for vertical interoperability: "Will the application run on my supercomputer?" Yes. "Will it run on my Unix workstation?" Yes. "Will it also run on my Macintosh or PC?" Yes. This approach has been shown to be a practical (and even preferable) software requirement. Vertical interoperability approaches are best supported by open nonproprietary specifications developed by responsive standardization groups such as the Internet Engineering Task Force (IETF).

Broad interoperability and Internet compatibility are essential. Closed solutions are dead ends. In order to achieve broad vertical integration, proprietary and vendor-specific hardware need to be avoided. Videoteleconferencing systems are an example of a market fragmented by competing and incompatible proprietary specifications. In the area of new connection-oriented cell-based services such as Asynchronous Transfer Mode (ATM) and Integrated Services Digital Network (ISDN), some disturbing trends are commonplace. Supposedly standardized protocol implementations often do not work as advertised, particularly when run between hardware from different vendors. Effective throughput is often far less than maximum bit rate. Latency performance is highly touted and rarely tested. Working applications that use these services are difficult to find. Corresponding network operating costs are often hidden or ignored. Perhaps worst of all, long-haul services such as ATM may not be fully interoperable with the Internet Protocol and some core functionality (such as many-to-many multicast) may not be feasible. Application developers are advised to plan and budget for lengthy delays and costly troubleshooting when working with new services.

Connectivity conclusions. We have seen that connectivity has many facets. A technical overview of these concepts was necessary in order to understand what is feasible. Our technical foundation is now strong enough to consider broader concepts of content, interaction, economics, applications and personal impacts.

3. Content and the World Wide Web. Connectivity isn't worth much unless there's information of value (content) to exchange. While human and artificial users have created a wide variety of content for many years, only recently has it been available on a global scale. Many proposals have been made regarding the optimum way to achieve a global database, but only one methodology is a contender: the Web.

If the Internet is our global supercomputer, then the Web is our global database. The Web has been defined as a "wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of

documents" [Hughes 94]. Fundamentally, the Web combines a name space consisting of all visible information stores on the Internet with a broad set of retrieval clients and servers, all of which can be connected by easily defined Hypertext Markup Language (HTML) hypermedia links. This globally accessible combination of media, client programs, servers and hyperlinks can be conveniently utilized by humans or autonomous entities. The Web continues to fundamentally shift the nature of information storage, access and retrieval, best explained by its originator [Berners-Lee 94]. Information content is the crucial component underlying most 3D graphics renderings. Information context is the key quality provided by the universal naming scheme and simplicity of HTML. Anyone can now add any content and any context they might want.

Current Web capabilities are easy to use despite rapid growth and change. An extension mechanism allows new content types and corresponding viewers to be easily integrated into existing browsers. The relative simplicity of HTML has been essential to the rapid growth of the Web. People add their own content because it's easy to do. Convention and common use have standardized many content formats, and ease of use has been the principal engine of success.

The ubiquitous availability of content via the Web has massive implications that we are only beginning to perceive. It's one thing to be aware that a Library of Congress exists somewhere, with most information sources that might ever be wanted. It's another to have such an information archive close at hand for easy access, regardless of your physical location [Library 95]. It's a further step to think that such an archive might be rapidly searchable and conveniently retrievable on demand. Until recently it was unprecedented to think that anyone anywhere might contribute to such a store without formal constraint. Now such capabilities are common. Future content challenges include easy connections for datasets, databases and analytic models to the Web.

These ideas are a series of comprehensible steps, but together they imply amazing capabilities. If knowledge is power, then these capabilities are powerful indeed. It is now possible for any question on nearly any subject to quickly provoke more information than is humanly possible to absorb. Eventually "everybody's everything" can be online. It seems there is now an ocean of information surrounding us. We can swim around the world or drown trying to drink it all in. How we deal with it is our choice.

While the full implications of universal access to massive content are still emerging, it's clear the genie can't be stuffed back into the bottle. We must deal with all this content. Such a challenge is a good match for interactive 3D computer graphics. Since graphics techniques can portray large amounts of information in a meaningful way, realistic rendering and scientific visualization techniques are logical choices for dealing with massive content. Effective 3D user interfaces are also needed for navigating the richness of these worlds. Interfaces must feel intuitive to both novice and expert, particularly if access and interaction are to scale up to match amazingly large volumes of content.

There are two more important points related to networking and the Web: improved interaction and importance of an open standards process. The Hypertext Transfer Protocol (http) which is used for most Web-based interactions typically provides a client-server relationship; a user can push on a Web resource and get a response, but there are no openly standardized mechanisms for a Web application to independently push back at the user. Netscape has independently defined and implemented server-push and client-pull techniques that permit repeated queries and replies [Netscape 95]. An example of "push" is when a server can send repeated updates down an open connection to refresh a client display. An example of "pull" is automatic repeated queries by a client at intervals matching data refresh cycles on the server. These methods for improving interaction on the Web are steps in the right direction. Better interaction mechanisms will enable new types of Web applications.

Netscape preempted the HTML standards process by unilaterally proposing and incorporating push/pull capabilities. Although server push and client pull are useful concepts likely to be included in forthcoming versions of the HTML specification, unilateral implementations can fragment a common standard into incompatible, inadequate variants. Open standards review means that any specification can be widely scrutinized in detail. Meanwhile (in the IETF at least) two or more independent implementations are needed to validate correct performance prior to acceptance as a standard. Patience and cooperation are essential since coopting standardization through market share creates far more bottlenecks than breakthroughs. Considering this particular technical example, as the demands of interactive client/server queries and replies grow in scale, many of the bandwidth and scaling issues explored in the MBone will reappear. Since cooperative standards development has been essential for the success of the MBone, similar cooperative standards-based efforts will continue to be essential as we scale up the Web.

4. Virtual Environments (VEs). The scope of virtual environment development is so broad that it can be seen as an inclusive superset of all other global information infrastructure applications. Virtual environments and virtual

reality applications are characterized by human operators interacting with dynamic world models of truly realistic sophistication and complexity [Zyda 93] [Durlach 95]. Current research in large-scale virtual environments can link hundreds of people and artificial agents with interactive 3D graphics, massive terrain databases, global hypermedia and scientific datasets. Related work on teleoperation of robots and devices in remote or hazardous locations further extends the capabilities of human-machine interaction in synthetic computer-generated environments. VE construction can include concepts and components from nearly any subject area. The variety of desired connections between people, artificial entities and information can be summarized by the slogan "connecting everyone to everything." As diversity and detail of virtual environments increase without bound, network requirements become the primary bottleneck.

The most noticeable characteristic of virtual environments is interactive 3D graphics, which ordinarily involves coordinating a handful of input devices while placing realistic renderings at fast frame rates on a single screen. Networking can connect virtual worlds with realistic distributed models and diverse inputs/outputs on a global scale. Graphics and virtual-world designers interested in large-scale interactions can now consider the world-wide Internet as a direct extension of their computer. A variety of networking techniques can be combined with traditional interactive 3D graphics to provide almost unlimited connectivity. Experience shows that the following services are essential for virtual world communications: reliable point-to-point communications, behavior interaction protocols such as the IEEE standard Distributed Interactive Simulation (DIS) protocol, World Wide Web (WWW) links, and multicast channels for bandwidth-efficient many-to-many communications. Further examination of behavior-based interaction protocols is next presented using DIS as an example methodology.

Distributed Interactive Simulation (DIS). The DIS protocol is an IEEE standard for logical communication among entities in distributed simulations [IEEE 93] [DIS 94]. Although initial development was driven by the needs of military users, the protocol formally specifies the communication of physical interactions by any type of physical entity and is adaptable for general use. Information is exchanged via protocol data units (PDUs) which are defined for a large number of interaction types.

The principal PDU type is the Entity State PDU. This PDU encapsulates the position and posture of a given entity at a given time, along with linear and angular velocities and accelerations. Special components of an entity (such as the orientation of movable parts) can also be included in the PDU as articulated parameters. A full set of identifying characteristics uniquely specifies the originating entity. Several complementary dead-reckoning algorithms permit computationally efficient projection of entity posture by listening hosts. Dozens of additional PDU types are defined for simulation management, sensor or weapon interaction, signals, radio communications, collision reporting and logistics support. Most of these highly specialized "garbage PDUs" are of questionable utility for most VE applications. What is essential is correct (or at least "close-enough") physics, real-time realism and adequate global consistency.

Here is an example showing how the DIS entity state PDU properly enables networked distribution of realistic physics-based behavior. In 1992 a jet pilot at the Naval Postgraduate School developed a real-time model of high-performance aircraft response using parameterized flight coefficients and quaternion mathematics [Cooke 92]. It turns out that tactical aircraft are inherently unstable. Jet pilots require years of training and onboard computers are continuously making compensatory adjustments to flight surfaces. Qualified fighter pilots are pleased with this quaternion-based model since the simulated jet reacts realistically and "flies" much like a true jet. Unfortunately, naive operators tend to crash when flying this model (and presumably might experience similar difficulties in a real jet). Substituting flight coefficients that represent a more benign aircraft like a Piper Cub made operation much easier for casual users. The significant fact related to this substitution is that the underlying DIS model requires no changes. Regardless of whether packet update rates necessary for smooth motion average once per 5 seconds or 20 times per second, the DIS Entity State PDU dead reckoning algorithms are able to properly track position and orientation. This maximizes precision, maintains proper global consistency, and minimizes network loading.

Of further interest to virtual-world designers is an open-format Message PDU. Message PDUs enable user-defined extensions to the DIS standard. Such flexibility coupled with the efficiency of Internet-wide multicast delivery permits extension of the object-oriented message-passing paradigm to a distributed system of essentially unlimited scale. Free-format DIS Message PDUs might provide general message-passing connectivity to any information site on the Internet, extended by use of network pointer mechanisms that already exist for the World Wide Web. This is a promising area for future work.

VE grand challenges. The most important "grand challenges" of computing today are not large, static gridded simulations such as computational fluid dynamics or finite element modeling. Similarly, traditional supercomputers are not necessarily the most powerful or significant platforms because adding hardware and dollars to incrementally

improve existing expensive computer designs is a well-understood exercise. What's more challenging and potentially more rewarding is the interconnection of all computers in ways that support global interaction of people and processes. In this respect, the Internet is the ultimate supercomputer, the Web is the ultimate database, and any networked hardware in the world is a potential input/output device. Large-scale virtual environments attempt to simultaneously connect many of these computing resources in order to recreate the functionality of the real world in meaningful ways. Network software is the key to solving VE grand challenges.

Large-scale virtual-world internetworking. Four key communication methods are necessary for large-scale virtual-world internetworking: light-weight messages, network pointers, heavy-weight objects and real-time streams (Figure 5). For each of these four methods, bandwidth and latency must be carefully considered. The following networking elements can be combined to implement these four methods.

- Distribution of virtual-world components using point-to-point sockets can be used for tight coupling and real-time response of physics-based models.
- DIS is a well-tested behavior protocol that enables efficient live interaction between multiple entities in multiple virtual worlds.
- A wide variety of alternative behavior protocols are being developed and evaluated for large-scale virtual-world interaction.
- Hypermedia servers and embedded Web browsers provide virtual worlds global access to pertinent archived images, video, graphics, papers, datasets, software, sound clips, text or any other computer-storable media, both as inputs and outputs.
- Multicast protocols permit moderately large real-time bandwidths to be efficiently shared by an unconstrained number of hosts.
- MBone connectivity permits live distribution of graphics, video, audio, DIS and other streams world-wide in real time.

Together these example components provide the core functionality of light-weight messages, network pointers, heavy-weight objects and real-time streams. Integrating these network tools in virtual worlds produces realistic, interactive and interconnected 3D graphics that can be simultaneously available anywhere [Brutzman 94] [Macedonia 95a, b].

Light-weight Interactions. Messages composed of state, event and control information as used in DIS Entity State PDUs. Implemented using multicast. Complete message semantics is included in a single packet encapsulation without fragmentation. Light-weight interactions are received completely or not at all.

Network Pointers. Light-weight network resource references, multicast to receiving groups. Can be cached so that repeated queries are answered by group members instead of servers. Pointers do not contain a complete object as light-weight interactions do, instead containing only a reference to an object.

Heavy-weight Objects. Large data objects requiring reliable connection-oriented transmission. Typically provided as an ftp or http response triggered by a network pointer request.

Real-time Streams. Live video, audio, DIS, 3D graphics images or other continuous stream traffic that requires real-time delivery, sequencing and synchronization. Implemented using multicast channels.

Figure 5. Four key communications components used in virtual environments.

Application layer interactivity. It is application layer networking that needs the greatest attention in preparing for the information infrastructure of the near future. DIS combined with multicast transport provides solutions for many application-to-application communications requirements. Nevertheless DIS is insufficiently broad and not adaptable enough to meet general virtual environment requirements. To date, most of the money spent on networked virtual environments has been by, for and about the military. Most remaining work has been in (poorly) networked games. Neither example is viable across the full spectrum of applications. There is a real danger that specialized high-end military applications and chaotic low-end game hacks will dominate entity interaction models. Such a situation might well prevent networked virtual environments from enjoying the sustainable and compatible exponential growth needed to keep pace with other cornerstones of the information infrastructure.

Next-generation behavior protocols. Successors to DIS are needed that are simpler, open, extensible and dynamically modifiable. DIS has proven capabilities in dealing with position and posture dead-reckoning updates,

physically based modeling, hostile entity interactions and variable latency over wide-area networks. DIS also has several difficulties: awkward extendability, requiring nontrivial computations to decipher bit patterns, and being a "big" (i.e. complicated) standard. DIS protocol development continues via a large and active standards community. However, the urgent military requirements driving the DIS standard remain narrower than general virtual environment networking requirements. Smaller and more efficient behavior protocols are being investigated by several researchers. The current NPS approach for next-generation behavior communications is to develop a "dial-a-protocol" capability, permitting dynamic modifications to the protocol specification to be transmitted to all hosts during a simulation [Stone 96]. A dynamically adjustable protocol is necessary for interactively testing global and local efficiency of distributed entity interactions. Our research is motivated by a fundamental networking principle: realistic testing and evaluation are essential, because initial performance of distributed applications never matches expectations or theory. Thus we're intentionally focusing on mechanisms for finding bottlenecks and demonstrating breakthroughs.

Other interaction models. Many other techniques for entity interaction are being investigated, although not always in relation to virtual environments. Several breakthroughs are possible. Just like in the real world, many ways of interacting will be valuable in virtual environments.

- Common Gateway Interface (cgi) scripts are executed on the queried host, often using Hypertext Transfer Protocol (http) [Berners-Lee 94] query extensions as inputs.
- Java has sparked wide interest as a network-based behavior language. Java enables simple and secure passing of precompiled program object files (applets) for compatible multiplatform execution [Sun 95].
- Intelligent agent interactions are an active area of research driven by the artificial intelligence and user interface communities. Rule-based agents typically communicate via a message-passing paradigm that is a straightforward extension of object-oriented programming methods.
- Instead of exchanging messages, mobile programs called agents can transport themselves. Telescript is a promising language designed for safe and secure mobile agent interactions [White 95].
- Ongoing research by the Linda project uses a generalized message-passing construct called "tuples" for logical entity interaction, with emphasis on scaling up to indefinitely large sizes and participants [Gelernter 92].
- MUDs (multi-user dungeons) and MOOs (MUDs Object-Oriented) provide a powerful server architecture and text-based interaction paradigm well suited to support a variety of virtual environment scenarios [Curtis 94].
- Passing interpretable scripts over the network in conjunction with predeployed, precompiled source code has been widely demonstrated for the multiplatform 2D graphical user interface (GUI) Tool Control Language (Tcl) [Ousterhout 94].

Virtual Reality Modeling Language (VRML). The Web is being extended to three spatial dimensions thanks to VRML, a specification originally based on an extended subset of the SGI OpenInventor scene description language [Wernicke 94]. Key contributions of the initial VRML 1.0 standard were a core set of object-oriented graphics constructs augmented by hypermedia links, all suitable for scene generation by browsers on Intel and Apple personal computers as well as Unix workstations. The interaction model for 3D VRML browsers remains client-server, similar to most other Web browsers. 3D browsers are usually embedded in 2D browsers or are launched as helper applications when connecting to a 3D site. VRML specification development has been effectively coordinated by mail list, enabling consensus by a large, active and open membership [Pesce 94, 95] [Bell 94].

Difficult issues regarding real-time animation in VRML 2.0 (Carey 96) include entity behaviors, user-entity interaction and entity coordination. Currently animation scripts and experimental servers are used for such functionality. In order to scale to many simultaneous users, peer-to-peer interactions will be necessary in addition to client-server query-response. Although "behaviors" are not formally specified, VRML 2.0 provides local and remote scripting language hooks (i.e. an applications programming interface or API) to graphical scene descriptions. Dynamic scene changes can thus be stimulated by scripted actions, message passing, user commands or behavior protocols, implemented using either Java or complete VRML scene replacement. A great deal of experimentation is in progress. It appears clear that successful global solutions for VRML behaviors are possible that provide simplicity, security, scalability, generality and open extensions. The most successful experimental approaches to shared behaviors will likely establish employment conventions and may even inspire behavior specifications. Finally, as the demanding bandwidth and latency requirements of virtual environments begin to be exercised by VRML, some client-server design assumptions of the Hypertext Transfer Protocol (http) may no longer be valid. Users won't be satisfied with network mechanisms that fail to accommodate high-bandwidth information streams or break down after a few hundred players. A Virtual Reality Transfer Protocol (vrtp) will be needed to take advantage of available transport-layer functionality and overcome bottlenecks in http. Experimentation and quantitative evaluation will be

essential to develop the next generation of code for diverse interentity virtual environment communications. To achieve these ambitious goals, it is necessary to create a Cyberspace Backbone (CBone) providing dedicated network resources for a virtual environment testbed.

Note that computer graphics applications now become more flexible than television. From a network perspective, prerendered computer graphics streams are equivalent to video streams. Behavior-based VRML will provide dynamic graphics scene delivery at relatively low bandwidths with viewpoints independently controlled by subscribing users. Effective deployment of globally networked graphics can complement television in fascinating ways.

- **5. Economics**. Predicting economic developments in new technologies is a risky business. The following points are only a few of many changes, emphasizing major new developments that are likely to produce market impacts. Users and investors can draw their own conclusions.
 - Marketplace providers may be far different from traditional companies. Individuals and small groups of
 people can be economic players, for example. GigaCorp Intergalactic UltraServices Ltd. might be one
 person working at home producing information products that are as good as (or better than) the products a
 large multinational corporation produces. Unrestricted access to information, open interest-group efforts and
 affordable high-quality software are key. These assets produce the new trump cards in creative product
 development: imagination, cooperation and rapid turnaround.
 - Value-added becomes a market basis. At first it might seem that when so much is free, no one needs to buy
 anything. In practice, people are willing to pay when there's significant value-added relative to what's free. For
 example, numerous software products are available today in identical free and supported versions.
 Nevertheless today's innovative value-added feature is tomorrow's commonplace free feature. Thus ongoing
 innovation, responsiveness and improvement are necessary for software program survival.
 - What is "free" in the computer software community is a sliding standard that tracks along with the exponential growth of the computer price/performance technology curve. Historically, companies charging for quality software have been first to market, with quality freeware versions following later. Programmers unwilling or unable to pay fees often create public domain software comparable to commercial versions. Current free availability of quality software development tools and code libraries has accelerated this growth of free software projects. Freeware programmers are often willing to port software to multiple architectures. Mailing lists, news groups, online hyperlinked documentation and software repositories encourage the growth of diverse and technically proficient user communities. High-quality online experience is usually available to open interest-group members (however informally) at all hours of the day or night. Open cross-platform public-domain (free) solutions frequently inspire profitable commercial software products. Both climb the cutting-edge-technology curve in tandem.
 - Information dissemination is easier than ever. Content for mass access is no longer solely controlled by television networks, radio broadcasters and print publishers. As more people put information online and collaborate with groupware, traditional information sources become less necessary. Just as television didn't put radio out of business, expect to see evolutionary changes in information sources rather than elimination of other media. Open access to disparate media over the network fosters both competition and cooperation. These are democratizing and equalizing forces. Everybody will want access to everything, and networking makes it possible. Interactive 3D graphics with independent viewpoint, navigation and content already provides serious competition to television. New 3D "channels" are creating new businesses, arts and entertainments.
 - It becomes increasingly difficult to preserve original sources as internetworked information grows
 exponentially. Digital libraries will become essential to provide continuity of access. Context may be
 supported through ever-more-sophisticated links, browsers and virtual environments, but secure storage of
 basic content will remain a fundamental requirement. A forward-looking overview of digital library efforts is
 given by [Garrett 95].
 - Informal market analysis of an industry that undergoes a complete reorganization every year or two indicates
 that open methods based on responsive standards survive, while closed proprietary methods struggle and
 die, or get absorbed/killed/subsumed by dominant companies in a market (like Microsoft). Semiopen
 methods (like ISO OSI) that are overconstrained by standardization or restrictive rules fall behind the pace of
 the marketplace and become obsolete in effect they appear proprietary.
 - Security considerations affect all parts of the Internet, including multiuser virtual environments. Encryption, authentication and nonrepudiation are all feasible using a variety of currently available open and closed cryptographic standards. Public key cryptography, next-generation IPv6 safeguards and de facto market standards/solutions used for credit card transactions appear to be the most significant technologies.
 Next-generation security must safeguard entire networks in addition to individual users, clients and servers.

- Security is a moving target that will be solved mostly through independent efforts that eventually gain acceptance on the Internet. Security validation for virtual environments will hinge on the effectiveness of these methods when used together with challenging high-bandwidth fast-response applications.
- Economic problems on both personal and corporate scales often result from misunderstanding or misinformation. This happens everywhere, including the computing and academic communities. Marketing hyperbole takes advantage of the dizzying effects of ceaseless rapid change and the individual's need to stay informed. Rapidly growing bodies of knowledge and information are now part of the intellectual marketplace. Good and bad untested ideas go into the market early hoping to attract resources and influence the development of related work. However, the bottom line is results. Implementation (can it be built?), test (what does it really do?) and evaluation (does it do what we want?) are truer measures of success. Implementation, test and evaluation are the forcing functions that will produce working large-scale internetworked graphics environments. Visible results cut through the inevitable fog of hype and distorted information that surrounds rapid technical progress.
- **6. Applications.** Working applications will drive progress, rather than theories or hype. In this section we present feasible applications that are exciting possibilities or existing works in progress. Many new projects are possible and likely in the near future if virtual environment requirements are adequately supported by the global information infrastructure. Several conjectured scenarios follow.

Sports: live 3D stadium with instrumented players. Imagine that all of the ballplayers in a sports stadium wear a small device that senses location (through the Global Positioning System and local spread-spectrum radio signal sensing) and transmits behavior packets over a wireless network. Similar sensors are embedded in gloves, balls, bats and even shoes. A computer server in the stadium feeds player telemetry into a physically based articulated human model, that in turn extrapolates individual body and limb motions. The server also maintains a scene database for the stadium complete with texture-mapped images of the edifice, current weather and representative pictures of fans in the stands. Meanwhile, Internet users have 3D browsers that can navigate and view the action from any perspective in the stadium. Users can also tune to multicast channels providing updated player positions and postures (via light-weight behavior interactions) along with live audio and video. Statistics, background information and multimedia home pages are available for each player. Some remote spectators might even choose to publicly share their own play-by-play commentary. Thus any number of remote fans might supplement traditional television coverage with a live interactive computer-generated view. Perhaps the most surprising aspect of this scenario is that all component software and hardware technologies exist today. A proposed personal tracking system (PTS) suitable for multiple players in an stadium environment is presented in Figure 6 [Bible 95].

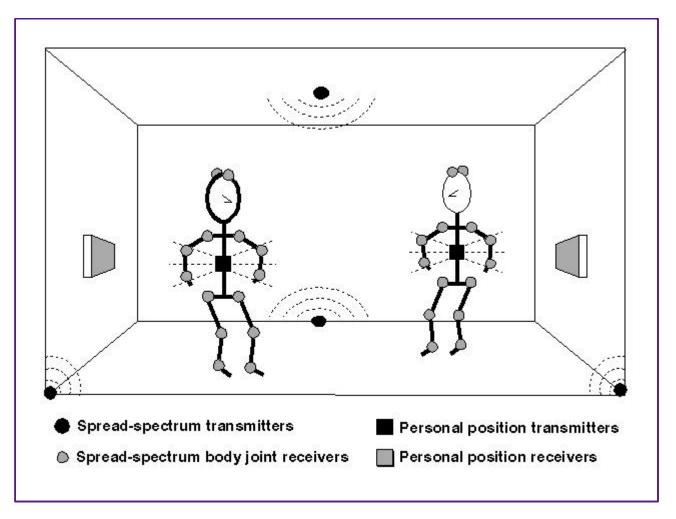


Figure 6. Example personal tracking system (PTS) using spread-spectrum radio ranging in a virtual environment [Bible 95]. Fourteen independent joint trackers passively determine body posture using a scheme similar to the Global Positioning System (GPS), then combined joint signals are multiplexed and transmitted back to the surrounding environment.

Education. Many people are trying to capitalize on the potential of global content by combining internetworking with education, long-distance teletraining and life-long learning. Applying computer graphics to education is also a steadily growing area of activity [Owen 95]. Our approach to learning and research includes grounding all theoretical concepts in real-world applications and events. Networked education efforts in the Monterey Bay region aim at enabling teachers and students of all ages to use all aspects of the Internet [Brutzman 95a, b] [Bigelow 95]. Since students study a great deal of content, regularly produce context and are excellent evaluators of effectiveness, it's likely that the best new examples combining innovative content with networked graphics will emerge from the educational community. Supporting educational network construction and teacher training is an excellent way to build a collaborative user base for new applications.

Military: 100,000 player problem. "Exploiting Reality with Multicast Groups" describes ground-breaking research on increasing the number of active entities within a virtual environment by several orders of magnitude [Macedonia 95a, b] Multicast addressing and the DIS protocol are used to logically partition network traffic according to spatial, temporal and functional entity classes as shown in Figure 7. "Exploiting Reality" further explains virtual environment network concepts and includes experimental results. This work has fundamentally changed the distributed simulation community, showing that very large numbers of live and simulated networked players in real-world exercises are feasible. The military origins of this problem will become irrelevant as our abilities to effectively interact socially, scientifically and educationally scale to the hundreds of thousands. Such growth from military to real-world problems is analogous to the original design of the ARPANET and military predecessors to today's Internet. Many different types of simulations can be connected via distributed virtual environments. An excellent reference that describes and organizes a thorough variety of analytic simulation approaches is [Fishwick 95].

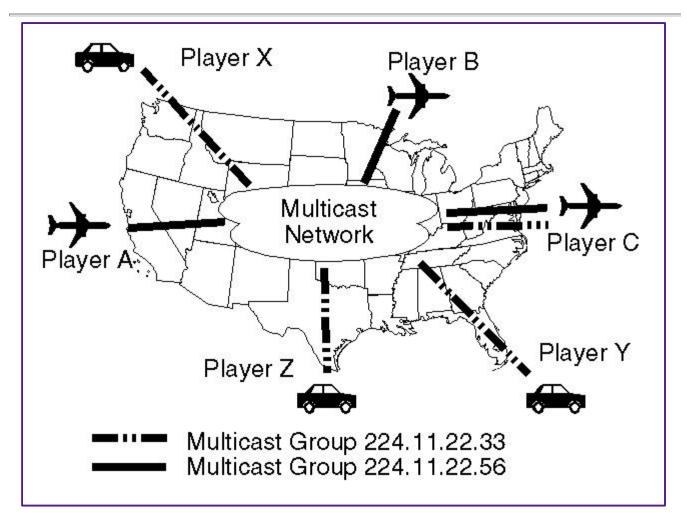


Figure 7. Multicast channels can partition network communications into spatial, temporal and functional classes to minimize traffic and exploit reality [Macedonia 95a, b].

Science: virtual worlds as experimental laboratories for robots and people. In separate work, we've shown how an underwater virtual world can comprehensively model all salient functional characteristics of the real world for an Autonomous Underwater Vehicle (AUV) in real time [Brutzman 94]. This virtual world is designed from the perspective of the robot, enabling realistic AUV evaluation and testing in the laboratory. In this context, our "Turing Test" for the virtual environment is whether the robot behaves identically in the underwater virtual world and in the real world. Real-time 3D computer graphics are our window into that virtual world (Figure 8). Visualization of robot interactions within a virtual world permits sophisticated analyses of robot performance that are otherwise unavailable. Sonar visualization permits researchers to accurately "look over the robot's shoulder" or even "see through the robot's eyes" to intuitively understand sensor-environment interactions. Theoretical derivation of six-degree-of-freedom hydrodynamics equations has provided a general physics-based model capable of replicating highly nonlinear (yet experimentally verifiable) response in real time. Distribution of underwater virtual-world components enables scalability and rapid response. Networking allows remote access, demonstrated via MBone audio and video collaboration with researchers at distant locations. Integrating the World Wide Web allows rapid access to resources distributed across the Internet. Distributed resources include a text-to-speech sound server that translates robot reports into sound files [Belinfante 94]. Ongoing challenges include scaling up the types of interactions, datasets, models and live streams that can be coordinated within the virtual world.

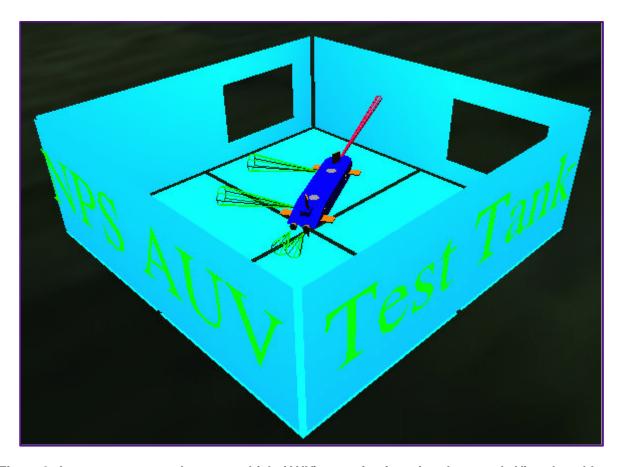


Figure 8. An autonomous underwater vehicle (AUV) operating in a virtual test tank. Virtual worlds can incorporate datasets, databases, models, robots and people [Brutzman 94].

Interaction: Multiple CAVEs using ATM and VRML. A CAVE is a walk-in synthetic environment that replaces the four walls of a room with rear-projection screens, all driven by real-time 3D computer graphics [Cruz-Neira 93]. These devices can accommodate 10-15 people comfortably and render high-resolution 3D stereo graphics at 15 Hz and higher update rates. The principal costs of a CAVE are in high-performance graphics hardware. We wish to demonstrate affordable linked CAVEs for remote group interaction. The basic idea is to send graphics streams from a master CAVE through a high-speed low-latency ATM link to a less expensive slave CAVE that contains only rear-projection screens.

Several serious ATM bottlenecks remain that must be resolved experimentally. Automatic generation of VRML scene graphs and simultaneous replication of state information over standard multicast links will permit both CAVEs and networked computers to interactively view results generated in real time by a supercomputer. Our initial application domain is a gridded virtual environment model of the oceanographic and biological characteristics of Chesapeake Bay, as shown in Figure 9 [Wheless 95, 96] [I-WAY 95]. To better incorporate networked sensors and agents into this and other virtual worlds, we are also investigating extensions to IP using underwater acoustics [Brutzman 95d]. As a final component, we are helping establish ambitious regional education and research networks which connect scientists, students from kindergartens through universities, libraries and the general public. Vertically integrated Web and MBone applications and the common theme of live networked environmental science are expected to provide many possible virtual-world connections [Brutzman 95a, b, c].

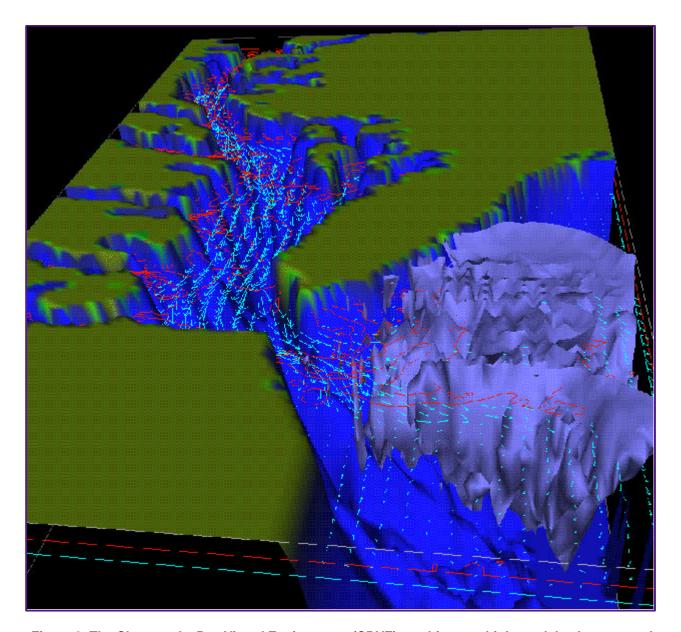


Figure 9. The Chesapeake Bay Virtual Environment (CBVE) combines multiple models, datasets and interaction modes with Sstandard IP and high-bandwidth ATM internetworking [Wheless 95, 96].

Mobile global public domain television. Access to the MBone means that anyone can become a global television provider. During SIGGRAPH 95 we experimented with mobile audio/video delivery to demonstrate location-independent multicasting. We loaded a cart with a camera-equipped SGI Indy workstation, a 2 Mbps wireless bridge, wireless microphones and video gear. Dubbed "MBone Unplugged," we were able to roam the convention floor while continuously multicasting exhibits and events world-wide (Figure 10) [Brutzman, Emswiler 95] [Clinger 96]. Working under conference auspices provides access to significant content with appropriate release permissions. We now believe that this rig is the workstation of the future: totally mobile, complete graphics and video capabilities, no apparent limits. Related work showed that live world-wide multicast of an academic course was possible for an entire quarter [Emswiler 95]. Planned future work includes simultaneous world-wide conference multicasting over ATM and MBone links while directly recording the audio/video at multiple bandwidths to a digital archive. We expect to show that previously transitory content produced by conference events can provide significant long-term value through easy, inexpensive live multicast and storage of high-bandwidth audiovisual streams.



Figure 10. "MBone Unplugged" cart used at SIGGRAPH 95 for wireless audio, video and 3D graphics with world-wide multicast scope [Brutzman, Emswiler 95].

7. Personal impacts. Tremendous enthusiasm and deep insights result when people use good interactive 3D graphics applications. Combining graphics and networks looks like a sure win, but the personal impacts of new technology are notoriously hard to predict. Here are a few ideas gained during our journey about how people will use large-scale internetworked virtual environments.

Large-scale means everyone and everything. This is a people process just as much as a technical process. Here are some people-related lessons learned from the ongoing growth of the Web. When something is easy, people do it. When something is challenging, people are willing to pay someone else or figure out another way to do it. When something is difficult, people don't do it. Examples of relatively easy things include building personal web pages and putting information online. Challenging things include professional web server providers and current web-server-in-a-box products. Excessively difficult things include early false starts by some commercial online services and overcomplex standardization efforts that never reach critical mass.

The personal impact of scaling up virtual environments to arbitrarily large sizes means that connecting and contributing to them can't be harder than setting up your own home page. If the process is not well defined, few people will risk wasting their time. As long as hooking up and authoring virtual worlds remain difficult, the technical solutions aren't yet good enough.

Information overload is the new norm. There's already too much information available. The unexpected will continue to happen, usually right when people think that they're caught up with what's current. The possibility of 500 television channels is ridiculously simple; there are already far more content sources on the Internet. With interactive opportunities increasing exponentially, personal impacts will likely have to be experienced to be understood. What does it mean when anyone can create, join, ignore or compete with any channel of information? No one really

knows but we'll undoubtedly find out. Some individuals look to science fiction for inspiration and models of behavior. Some virtual reality architects on the VRML mailing list appear to go beyond inspiration and use Snow Crash [Stephenson 92] as a design document. Life imitates art, science imitates science fiction.

8. Projections. Merely by reading the New York Times, any individual can get more information about the world than was available to any world leader throughout most of human history. Multiply that single stream by the millions of other information sources on the Internet, and it's clear we do not lack content. Mountains of content are accessible. What we need more is context, i.e. a way to interactively locate, retrieve and display related pieces of information in a timely manner. Context establishes the reliability and pedigree of information streams. It also permits selectively culling large volumes of unstructured information that can otherwise overwhelm any single person or process.

Within two lifetimes we have invented several new technologies for recording and exchanging information. Handwriting gave way to typing, then typing to word processing, and soon afterward came desktop publishing. Now people can use 3D real-time interactive graphics simulations and dynamic "documents" with multimedia hooks to record and communicate information. Such "documents" can be directly distributed on demand to anyone connected to the Internet. In virtual environments a further paradigm shift becomes possible. The long-term potential of virtual environments is to serve as an archive and interaction medium, combining massive and dissimilar datasets and data streams of every conceivable type. Virtual environments will enable comprehensive and consistent interaction within those massive datasets, data streams and models that recreate reality. Virtual environments can provide meaningful context to the mountains of content that currently exist in isolation without roads, links or order.

What about scaling up? Fortunately there already exists a model for these growing mountains of information content: the real world. Virtual worlds address the context issue by providing information links similar to those in the real world. When our virtual constructs cumulatively approach realistic levels of depth and sophistication, our understanding of the real world will deepen correspondingly. In support of this goal, we have shown how the structure and scope of virtual environment relationships can be dynamically extended using feasible network communications methods. This efficient distribution of information will let any remote user or entity in a virtual environment participate and interact in increasingly meaningful ways.

Open access to any type of live or archived information resource is becoming available for everyday use by individuals, programs, collaborative groups and even robots. Virtual environments are a natural way to provide order and context to these massive amounts of information. World-wide collaboration works, for both people and machines. Finally, the network is more than a computer, and even more than your computer. The Internet becomes our computer as we learn how to share resources, collaborate and interact on a global scale.

9. Conclusions. Interactive 3D computer graphics enables people to see things that are otherwise imperceptible, impractical or impossible. Computer networking can connect everyone to everything, any person to any machine to any information resource. Networks are not "something else" relative to graphics, rather they are integral as we scale up in every direction. As these interrelated technologies combine, understanding why limits exist will help us break through bottlenecks. Each bottleneck broken reveals another as we consider new-found capabilities and their corresponding limitations.

Large-scale virtual environments are the new grand challenge in computing, a challenge that can include every person and every subject. The best motivation for scaling up can be found in the respective strengths of 3D graphics and networking: perception and bandwidth. Interactive graphics let us perceive the world in new and familiar ways. Networks provide information bandwidth, broad connectivity and varied content at rapid rates. Our new challenges are fascinating. Combining graphics and networks now lets us work on perceptual bandwidth, maximizing the depth and breadth and speed we use to interpret the world around us. This achievement may be our biggest breakthrough.

References.

Belinfante, Axel, "Say..." text to speech sound server, University of Twente, Netherlands, 1994. Available at http://wwwtios.cs.utwente.nl/say/?

Bell, Gavin, Parisi, Anthony and Pesce, Mark, "The Virtual Reality Modeling Language (VRML) Version 1.0 Specification," 26 May 1995. Available via the VRML Repository at http://www.sdsc.edu/vrml

Berners-Lee, Tim, Cailliau, Luotonen, Ari, Nielsen, Henrik Frystyk, and Secret, Arthur, "The World Wide Web," *Communications of the ACM*, vol. 37 no. 8, August 1994, pp. 76-82.

Bible, Steven R., Zyda, Michael and Brutzman, Don, "Using Spread-Spectrum Ranging Techniques for Position Tracking in a Virtual Environment," *IEEE Networked Realities Workshop*, Boston Massachusetts, October 26-28 1995. Available at http://www-npsnet.cs.nps.navy.mil/npsnet/publications/NR95-Paper-Bible.ps.Z

Bigelow, Randall J., *Internetworking: Planning and Implementing a Wide-Area Network (WAN) for K-12 Schools*, Master's Thesis, Naval Postgraduate School, Monterey California, June 95. Available at http://www.stl.nps.navy.mil/~rjbigelo/thesis.html

Bradner, Scott O. and Mankin, Allison, editors, *IPng: Internet Protocol Next Generation*, Addison-Wesley, Reading Massachusetts, 1996. Additional information available at http://playground.sun.com/pub/ipng/html/ipng-main.html

Brutzman, Donald P., *A Virtual World for an Autonomous Underwater Vehicle*, Ph.D. Dissertation, Naval Postgraduate School, Monterey California, December 1994. Available at http://web.nps.navy.mil/~brutzman/dissertation/

Brutzman, Don, "Networked Ocean Science Research and Education, Monterey Bay California," *Proceedings of International Networking (INET) 95 Conference*, Internet Society, Honolulu Hawaii, June 27-30 1995. Available at http://www.isoc.org/HMP/PAPER/039/abst.html

Brutzman, Don, "Remote Collaboration with Monterey Bay Educators," *Visual Proceedings, Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics (SIGGRAPH) 95*, Los Angeles California, August 7-11 1995, p. 145.

Brutzman, Don and Emswiler, Tracey, "MBone Unplugged," SIGGRAPH 95 exhibit report, August 1995. Available at http://web.nps.navy.mil/~brutzman/unplugged.html

Brutzman, Don and Reimers, Stephen, "Internet Protocol over Seawater: Towards Interoperable Underwater Networks," *Unmanned Untethered Submersibles Technology 95*, Northeastern University, Nahant Massachusetts, September 25-27 1995, pp. 444-457. Available at ftp://taurus.cs.nps.navy.mil/pub/auv/ipoversw.ps

Carey, Rikk, Marrin, Chris and Bell, Gavin, editors, "The Virtual Reality Modeling Language (VRML) Version 2.0 Specification," International Standards Organization/International Electrotechnical Commission (ISO/IEC) draft standard 14772, August 4 1996. Available via the VRML Repository at http://www.sdsc.edu/vrml

Clinger, Marke, "GraphicsNet '95: Integrated Voice, Video, Graphics and Data Network Using Asynchronous Transfer Mode (ATM)," *COMPUTER GRAPHICS*, ACM SIGGRAPH, vol. 30. no. 1, February 1996, pp. 10-18. Additional information at http://www.siggraph.org/conferences/siggraph95/GraphicsNet

Cogger, Dick, CU-SeeMe, desktop videoconferencing software, Cornell University, Ithaca New York, 1995. Available at http://gated.cornell.edu/pub/video/html/Welcome.html

Comer, Douglas E., *Internetworking with TCP/IP Volume I: Principles, Protocols and Architecture*, second edition, Prentice Hall, Englewood Cliffs New Jersey, 1991.

Cooke, J.C., Zyda, M.J., Pratt, D.R. and McGhee, R.B., "NPSNET: Flight Simulation Dynamic Modeling using Quaternions," *PRESENCE: Teleoperations and Virtual Environments*, vol. 1 no. 4, Fall 1992, pp. 404-420. Available at

http://www-npsnet.cs.nps.navv.mil/npsnet/publications/NPSNET.Flight.Simulation.Dvnamic.Modeling.Using.Quatern

Cruz-Neira, Carolina, Leigh, Jason, Papka, Michael, Barnes, Craig, Cohen, Steven M., Das, Sumit, Engelmann, Roger, Hudson, Randy, Roy, Trina, Siegel, Lewis, Vasilakis, Christina, DeFanti, Thomas A. and Sandin, Daniel J., "Scientists in Wonderland: A Report on Visualization Applications in the CAVE Virtual Reality Environment," *IEEE* 1993 Symposium on Research Frontiers in Virtual Reality, San Jose California, October 25-26 1993, pp. 59-66 and CP-3.

Curtis, Pavel and Nichols, David A., "MUDs Grow Up: Social Virtual Reality in the Real World," *Proceedings of the IEEE Computer Conference*, IEEE Computer Society Press, Los Alamitos California, 1994, pp. 193-200. Available

at ftp://ftp.parc.xerox.com/pub/MOO/papers/MUDsGrowUp.ps

Deering, Steve, "Host Extensions for IP Multicasting," Request for Comments (RFC) 1112, Internet Engineering Task Force (IETF), August 1989. Available at ftp://ds.internic.net/rfc/rfc1112.txt

DIS Steering Committee, *The DIS Vision: A Map to the Future of Distributed Simulation*, version 1, May 1994. Available at *ftp://sc.ist.ucf.edu/public/STDS/vision.ps*

Durlach, Nathaniel I. and Mavor, Anne S., editors, *Virtual Reality: Scientific and Technological Challenges*, National Research Council, National Academy Press, Washington DC, 1995.

Emswiler, Tracey, Internetworking: Using the Multicast Backbone (MBone) for Distance Learning, Master's Thesis, Naval Postgraduate School, Monterey California, September 95. Summary video available at http://www.stl.nps.navy.mil/~iirg/emswiler/emswiler.qt.Z

Fishwick, Paul A., Simulation Model Design and Execution: Building Digital Worlds, Prentice Hall, Englewood Cliffs New Jersey, 1995. Additional information and SimPack simulation toolkit software available at http://www.cis.ufl.edu/~fishwick/

Garrett, John and Waters, Donald, cochairs, "Preserving Digital Information," Task Force on Archiving of Digital Information, Commission on Preservation & Access and the Research Libraries Group, draft report version 1.0, Yale University, New Haven Connecticut, August 24 1995.

Gelernter, David, Mirror Worlds - or the Day Software Puts the Universe in a Shoebox... How It Will Happen and What It Will Mean, Oxford University Press, New York, 1992.

Hughes, Kevin, "Entering the World Wide Web (WWW): A Guide to Cyberspace," Enterprise Integration Technology Inc., May 1994. Available at http://www.eit.com/web/www.guide/

IEEE Standard for Information Technology - Protocols for Distributed Interactive Simulation (DIS) Applications, version 2.0, Institute for Simulation and Training report IST-CR-93-15, University of Central Florida, Orlando Florida, May 28 1993.

International Wide-Area Year (I-WAY) project, North American ATM network in support of *IEEE/ACM Supercomputing 95*, San Diego California, December 3-7 1995. Information available at http://www.iway.org

Internet Network Information Center (Internic), Request For Comments (RFC) archive, <u>ftp://ds.internic.net</u>, 1995. Hypertext interface and search facilities available at <u>http://www.internic.net</u>

Library of Congress, home page, November 1995. Available at http://www.loc.gov/

Macedonia, Michael R. and Brutzman, Donald P., "MBone Provides Audio and Video Across the Internet," *IEEE COMPUTER*, April 1994, pp. 30-36. Available at ftp://taurus.cs.nps.navy.mil/pub/i3la/mbone.html

Macedonia, Michael R., *A Network Software Architecture for Large Scale Virtual Environments*, Ph.D. Dissertation, Naval Postgraduate School, Monterey California, June 1995. Available at http://www.cs.nps.navy.mil/research/npsnet/publications/Michael.Macedonia.thesis.ps.Z

Macedonia, Michael R., Zyda, Michael J., Pratt, David R., Brutzman, Donald P. and Barham, Paul T., "Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments," *IEEE Computer Graphics and Applications*, vol. 15 no. 5, September 1995, pp. 38-45. Available at http://www.cs.nps.navv.mil/research/npsnet/publications/IEEECGA.ps.Z

Malamud, Carl, Stacks: Interoperability in Today's Computer Networks, Prentice Hall, Englewood Cliffs New Jersey, 1992.

Netscape Corporation, "An Exploration of Dynamic Documents," online documentation page, November 1995. Available at http://www.netscape.com/assist/net_sites/pushpull.html

NRENAISSANCE Committee, Realizing the Information Future: The Internet and Beyond, Computer Science and

Telecommunications Board, National Research Council, National Academy Press, Washington DC, 1994. Available at http://www.nap.edu/nap/online/rtif/

Ousterhout, John K., Tcl and the Tk Toolkit, Addison-Wesley, Reading Massachusetts, 1994.

Owen, Scott, guest editor, focus issue on computer graphics education, *COMPUTER GRAPHICS, ACM SIGGRAPH*, vol. 29 no. 3, August 1995.

Pesce, Mark and Behlendorf, Brian, moderators, "Virtual Reality Modeling Language (VRML)," working group mail list, 1994-1996. Archived at http://vrml.wired.com/ and http://vrml.wired.

Pesce, Mark, VRML - Browsing and Building Cyberspace, New Riders Publishing, Indianapolis Indiana, 1995.

Schulzrinne, Henning, Casner, Stephen, Frederick, Ron and Jacobson, Van, "RTP: A Transport Protocol for Real-Time Applications," Request for Comments (RFC) 1889, Internet Engineering Task Force (IETF), January 1996. Available at ftp://ds.internic.net/rfc/rfc1889.txt

Stallings, William, Data and Computer Communications, fourth edition, Macmillan Publishing, New York, 1994.

Stephenson, Neal, Snow Crash, Bantam Books, New York, 1992.

Stevens, Richard W., UNIX Network Programming, PTR Prentice Hall, Englewood Cliffs New Jersey, 1990.

Stone, Steven, NPSNET: A Rapidly Reconfigurable Application-Layer Virtual Environment Network Protocol, Master's Thesis, Naval Postgraduate School, Monterey California, September 1996.

Sun Microsystems Corporation, Java language home page, November 1995. Available at http://java.sun.com/

Wernicke, Josie, The Inventor Mentor: Programming Object-Oriented 3D Graphics with OpenInventorTM, Release 2, Addison-Wesley Publishing, Reading Massachusetts, 1994. Summary available at http://www.sgi.com/Technology/Inventor/VRML/TIMSummary.html

Wheless, Glen, LaScara, Cathy, Valle-Levison, Arnaldo, Brutzman, Don and Sherman, Bill, "Chesapeake Bay Virtual Ecosystem Model (CBVEM): Interacting with a Couple Bio-Physical Simulation," *IEEE/ACM Supercomputing* 95, San Diego California, December 3-7 1995. Information available at http://www.ccpo.odu.edu/~wheless

Wheless, Glen H., LaScara, Cathy M., Valle-Levison, Arnaldo, Brutzman, Donald P., Sherman, William, Hibbard, William L. and Paul, Brian E., "Chesapeake Bay Virtual Ecosystem Model (CBVEM): initial results from the prototypical system," *International Journal of Supercomputer Applications*, to appear. Available at ftp://ftp.ccpo.odu.edu/pub/wheless/wheless ijsa.tar

White, James E., Telescript Technology: Mobile Agents, white paper, General Magic Inc., Sunnyvale California, 1995. To appear in Bradshaw, Jeffrey, ed., *Software Agents*, AAAI Press/MIT Press, Menlo Park California, 1996.

Zyda, Michael J., Pratt, David R., Falby, John S., Barham, Paul T., Lombardo, Chuck and Kelleher, Kristen M., "The Software Required for the Computer Generation of Virtual Environments," *PRESENCE: Teleoperators and Virtual Environments*, vol. 2 no. 2, MIT Press, Cambridge Massachusetts, Spring 1993, pp. 130-140. Available at http://www-npsnet.cs.nps.navy.mil/npsnet/publications/NPSNET.Software.Required.for.the.Computer.Generation.of.

Acknowledgements. I gratefully acknowledge contributions and insights gained from discussions with Mike Macedonia, Mike Zyda, Glen Wheless, Clark Dodsworth and members of the Virtual Reality Modeling Language (VRML) mail list. Funding support for this research was provided by the Naval Postgraduate School and the Office of Naval Research.

Author Information. Don Brutzman is a computer scientist working in the Interdisciplinary Academic Group at the Naval Postgraduate School. His research interests include underwater robotics, real-time 3D computer graphics, artificial intelligence, high-performance networking and large-scale virtual environments. He is a member of the Institute of Electrical and Electronic Engineers (IEEE), the Association for Computing Machinery (ACM) Special Interest Group on Graphics (SIGGRAPH), the American Association for Artificial Intelligence (AAAI), the Marine Technology Society (MTS) and the Internet Society (ISOC). Collaborative project information is available at

http://web.nps.navy.mil/~brutzman

The Uniform Resource Locator (URL) for this paper is http://web.nps.navy.mil/~brutzman/vrml/breakthroughs.html Your comments are welcome.

Maintainer: <u>Don Brutzman (brutzman@nps.navy.mil)</u>
Completed: 15 August 1997 Web links updated: 2 April 2001





Internetworked Graphics and the Web

Theresa-Marie Rhyne, Lockheed Martin Don Brutzman, Naval Postgraduate School Michael Macedonia, Fraunhofer Center for Research in Computer Graphics

lthough the networking and computer graphics fields are considered to be distinct disciplines, they must begin to converge in order to support collaborative exploration and information visualization on the Internet and the World Wide Web. Telecommunication breakthroughs remove bottlenecks and provide new opportunities for interactive 3D graphics across globally interconnected, dissimilar networks. Multicast Backbone (MBone) tools, developed in the networking arena, provide desktop videoconferencing tools for sharing information visualization and virtual reality explorations. The Virtual Reality Modeling Language (VRML), developed in the computer graphics arena, supports the 3D display and fly-through of networked computing resources on the Internet.

The computer graphics community considers VRML to be an interactive tool for exploring content on the Web. The telecommunications community calls it an application on the networking infrastructure. Here we define the concept of

Internetworked Graphics to describe the future merger and dependencies of computer graphics applications and the telecommunications networking infrastructure.

The concept of Internetworked Graphics can be examined from six perspectives:

- Connectivity: The capacity, bandwidth, protocols, and multicasting capabilities on the networking infrastructure.
- Content: Defined in terms of the Web, content encompasses any type of information, dataset, or stream used in the computer graphics environment.
- Interaction: Requires minimal latency, a sense of presence, and the ability to both access and modify content.
- Economics: Creation of financial mechanisms to support Internet and Web usage as well as online secure money transactions.
- Applications: Establish the context for internetworked graphics and determine infrastructure development.

 Personal aspects: The sense of wonder associated with information discovery and exploration.

For more information about these six aspects, see the article by Don Brutzman, "Graphics Internetworking: Bottlenecks and Breakthroughs," (*Digital Illusion*, Addison Wesley, 1997, to appear; http://www.stl.nps.navy.mil/~brutzman/breakthroughs.html).

EXPANDING INTO 3D WITH VRML

VRML has already extended the Web to three spatial dimensions. The VRML specification is based on an extended subset of the Silicon Graphics, Inc.'s OpenInventor scene-description language. Key contributions of the initial VRML 1.0 standard (available at http://www.sdsc.edu/vrml/spec.html) were a core set of object-oriented constructs augmented by hypermedia links.

Networking and computer graphics must converge to support collaborative exploration and information visualization on the Web.

VRML 1.0 allowed for scene generation by Web browsers on PCs as well as Unix workstations. The interaction model of 3D VRML browsers is client-server, similar to most other Web browsers. 3D browsers are usually embedded into 2D browsers (such as Netscape's Navigator or Microsoft's Internet Explorer) or launched as helper applications when connecting to a 3D site.

VRML 2.0, released in August 1996 (http://www.sdsc.edu/vrml/spec.html), expands VRML 1.0 to address real-time animation issues on the Web. VRML 2.0 provides local and remote hooks (APIs) to graphical scene description. It enables the simulation of dynamic scene changes by any combination of scripted actions, message passing, user commands, or behavior protocols. In order to scale to

many simultaneous users, VRML 2.0 requires peer-to-peer interactions in addition to client-server's query-response model. Here we begin to hit upon one of the bottlenecks of Internetworked Graphics. Future networked VRML scenes will demand significant real-time streaming in addition to client-server interactions.

The development of the next generation of Web software requires effective implementation of these Internetworked Graphics concepts.

Rendering Internetworked Graphics is a matter of human interface technology. VRML primitives provide the raw material for anyone to put together a sophisticated 3D scene that is viewable on any computer platform. VRML 2.0 script connections also enable scene animation, locally and globally. However, the Internet's applications layer does not provide sufficient support for VRML demands about to be made by users and content developers who will start connecting every imaginable geometry to every imaginable information stream. A new protocol is needed to extend the Web into fully interactive 3D large-scale virtual environments. This protocol will need to address the integrated needs of computer graphics content developers with the telecommunications infrastructure.

MBONE FOR VIRTUAL COMMUNICATION

The MBone is one of the Internet's most interesting capabilities (see M.R. Macedonia and D.P. Brutzman, "MBone Provides Audio and Video Across the Internet," *Computer*, Apr. 1994, pp. 30-36). It is used for live audio, live video, and packets such as Distributed Interaction Simulation (DIS) on a global scale.

MBone is a virtual network—it shares the same physical media as the Internet and relies on a network of routers (*mrouters*) that can support multicast. Mrouters are either upgraded commercial routers or dedicated workstations that run modified operating system kernels in parallel with standard routers.

Two things make multicasting feasible on a global scale: the installation of high-bandwidth Internet backbone connections and the widespread availability of workstations with adequate processing power and built-in audio capability.

Multicast provides one-to-many, several-to-many, and many-to-many network delivery services. An example of a one-to-many service is the live transmission of a technical conference. A several-to-many service might be a panel interview among different locations attended by dispersed sets of people. A many-to-many service might be a distributed research exercise with 100 active contributors. Multicast is useful for a variety of videoconferencing, audio, and multiplayer events in which numerous hosts must communicate simultaneously.

Multicast currently uses only the User Datagram Protocol (UDP) and not the Transmisssion Control Protocol (TCP) of the Internet Protocol (IP) suite. Therefore, multicast streams are a connectionless, "unreliable" service: Lost MBone packets stay lost. This "best-effort" service requires no setup, uses no acknowledgments, and guarantees no delivery. Ordinarily, this is a good thing for most real-time information streams (such as audio and video) because it avoids delivery bottlenecks and unwanted overhead. However, this can pose difficulties for future development of collaborative 3D large-scale virtual worlds. Recently, Web tool developers have also begun to address incorporating desktop videoconferencing capabilities into Web browsers. A more carefully engineered backbone service that includes dedicated multicast capabilities, dedicated bandwidth, and dedicated latency is needed to support the future development of Web-based, 3D, large-scale virtual worlds.

VRML techniques have also recently been applied to visualizing the complexity of MBone virtual networks and Internetworked Graphics roadblocks (T. Munzner et al., "Visualizing the Global Topology of the Mbone," *Proc. IEEE Symp. Information Visualization*, IEEE

CS Press, 1996). Information visualization techniques will aid the future development of telecommunications technology.

BOTTLENECKS AND ROADBLOCKS

The development of the next generation of Web software requires effective implementation of these Internetworked Graphics concepts. Those who develop networking protocols and MBone capabilities must factor in the impact that VRML and interactive visualization will have on future bandwidth and latency. Those who create Web-based interactive graphics tools must understand the underlying networking standards and infrastructure in order to build effective 3D collaborative applications. Without an integrated approach, bottlenecks and roadblocks will increase.

The evolution of commodity 3D computer graphics and visualization applications for PCs could present one roadblock. These applications depend on 3D graphics libraries for rendering and displaying images, and they accomplish real-time image display via accelerator cards or (potentially slower) software renderers. Developers who use these tools to create interactive 3D renderings and images to enable, for example, real-time data mining on the Internet will have to understand telecommunication bandwidth and protocols.

A key contribution of research in this area would be development tools, protocols, and algorithms that abstract away the complexity of Internetworked Graphics. These tools would provide mechanisms for effectively managing bandwidth, latency, and reliability automatically, depending on the developer's intent.

PERSONAL ASPECTS

The positive personal aspects associated with Internetworked Graphics will center on the ability to participate in real-time collaborative information discovery and visual exploration among geographically remote users.

The negative aspects will be associated with future Web development efforts. Setting up and learning to use the networking and Web infrastructure can be all-consuming and thus refocus the basic

information sharing and education process. Also, the display capabilities of desktop PCs and workstations differ significantly. Even with the best networking infrastructure, some VRML worlds experienced through low-end computers will not have the intended interactivity or display characteristics. These difficulties will ultimately cause discomfort and frustration for Internetworked Graphics content developers and users. Only with an interchange between the networking and graphics communities of developers can we achieve potential reduction in these Web bottlenecks and roadblocks.

e are presently developing tutorials that teach the networking community about 3D computer graphics and the graphics community about telecommunications issues (http://siggraph.org/~rhyne/com97/com97-tut.html).

We have also developed proposals for a virtual reality transfer protocol (http://www.stl.nps.navy.mil/~brutzman/vrtp/). VRTP is intended to be publicdomain software that will provide any desktop computer with client, server, peer-to-peer, and network monitoring capabilities. People will use VRTP with 3D Internet browsers to navigate and join large, interactive, fully internetworked computer graphics environments. ❖

Theresa-Marie Rhyne is a lead scientific visualization researcher for Lockheed Martin at the United States Environmental Protection Agency's Scientific Visualization Center; contact her at trhyne@vislab.epa.gov.

Don Brutzman is an assistant professor in applied science at the Naval Postgraduate School; contact him at brutzman@nps.navy.mil.

Michael Macedonia is vice president in charge of developing global virtual environments at Fraunhofer Center for Research in Computer Graphics, Inc.; contact him at mmacedon@crcg.edu.

Call for *Computer* Reviewers

.....

We seek volunteers to review manuscript submissions. For more information, send curriculum vitae and e-mail address to Michelle Saewert at m.saewert@computer.org. We publish the names of reviewers at the end of each year and write commendation letters for outstanding reviews. We need people with expertise in a variety of fields, including:

Internet

• All aspects of Internet development and use in a development environment, including Web development, intranets, and collaborative work.

Software

- Programming languages and techniques
- Software engineering
- Operating systems

Information systems

- Database management
- Information storage and retrieval (content analysis and indexing, information search and retrieval, online information services, library automation)
- Information systems applications (types of systems)

Information interfaces and presentation (multimedia information systems, user interfaces)

Data

- Data encryption
- Coding and information theory

Computer systems organization

- Processor architectures
- Computer-communication networks (network architecture and design, network protocols, network operations, distributed systems)

Hardware

- Memory structures
- Data communications
- Logic design
- Integrated circuits

Theory of computation

- Models and modes of computation
- Logics and meanings of programs
- Mathematical logic and formal languages

Computing methodologies

- Artificial intelligence
- Computer graphics
- Image processing
- · Pattern recognition
- Simulations and modeling

Computing milieux

- Computers and education
- Computers and society
- Legal aspects of computing
- Management of computing and information systems

.....

The Virtual Reality Modeling Language and Java

Don Brutzman Code UW/Br, Naval Postgraduate School Monterey California 93943-5000 USA brutzman@nps.navy.mil

Communications of the ACM, vol. 41 no. 6, June 1998, pp. 57-64. http://www.web3D.org/WorkingGroups/vrtp/docs/vrmljava.pdf

Abstract. The Virtual Reality Modeling Language (VRML) and Java provide a standardized, portable and platform-independent way to render dynamic, interactive 3D scenes across the Internet. Integrating two powerful and portable software languages provides interactive 3D graphics plus complete programming capabilities plus network access. Intended for programmers and scene authors, this paper provides a VRML overview, synopsizes the open development history of the specification, provides a condensed summary of VRML 3D graphics nodes and scene graph topology, describes how Java interacts with VRML through detailed examples, and examines a variety of VRML/Java future developments.

Overview. The Web is being extended to three spatial dimensions thanks to VRML, a dynamic 3D scene description language that can include embedded behaviors and camera animation. A rich set of graphics primitives provides a common-denominator file format which can be used to describe a wide variety of 3D scenes and objects. The VRML specification is now an International Standards Organization (ISO) specification (VRML 97).

Why VRML and Java together? Over twenty million VRML browsers have shipped with Web browsers, making interactive 3D graphics suddenly available for any desktop. Java adds complete programming capabilities plus network access, making VRML fully functional and portable. This is a powerful new combination, especially as ongoing research shows that VRML plus Java provide extensive support for building large-scale virtual environments (LSVEs). This paper provides historical background, a detailed overview of VRML 3D graphics, example VRML-Java test programs, and a look ahead at future work.

Development history. "Birds of a feather" sessions organized by Mark Pesce and Tony Parisi at the 1994 World Wide Web and SIGGRAPH conferences led to the formation of an open working group to examine potential technologies for a "virtual reality markup language," later changing markup to modeling. The initial goal of this effort was selection of a general 3D scene-description file format that was suitable for modification by addition of hyperlink semantics similar to the HyperText Markup Language (HTML). Open nomination and discussion of seven possible candidates led to selection of an extended subset of Silicon Graphics Inc. (SGI) *OpenInventor* file format [Wernicke 94] [Carey, Strauss 92]. Extensive online proposals and discussions punctuated by reported results became the hallmark of the VRML development process. Key contributions of the initial VRML 1.0 standard were selection of a core set of object-oriented graphics constructs augmented by hypermedia links, all suitable for geometry rendering by Web browsers on personal computers or workstations.

Language extensions for VRML. Although spirited debate and rough consensus successfully delivered an initial specification, there were two major limitations in VRML 1.0: lack of support for dynamic scene animation, and no traditional programming language constructs. Difficult issues regarding real-time animation in VRML 1.0 included entity behaviors, user-entity interaction and entity coordination. VRML 2.0 development tackled these issues directly, using event-driven ROUTEs to connect 3D nodes and fields to behavior-driven sensors and timing. "Language wars" were avoided by allowing other programming languages to communicate with the VRML scene via a Script node. Initial languages chosen are JavaScript for light-weight in-line calculations, and Java for full-fledged programming and network access. If Java or JavaScript are supported in a VRML browser, they must conform to the formal interface specified in (VRML 97) Annexes B and C, respectively. Major browsers now support both. Ongoing development of VRML continues via open working groups supported by the nonprofit VRML Consortium (VRMLC 97).

Behaviors. The term "behaviors" refers to making changes in the structure or appearance of a 3D scene. Thus a broad definition of a VRML behavior might be "any change in the nodes or fields of a VRML scene graph." VRML 97 provides local key-frame animation mechanisms and remote scripting language hooks (i.e. an applications

programming interface or API) to any scene graph component. Dynamic scene changes can be stimulated by scripted actions, message passing, user commands or behavior protocols, implemented using either via Java calls or complete VRML scene replacement. General approaches for VRML behaviors and scene animation are possible that provide simplicity, security, scalability, generality and open extensions. Using Java is the most powerful way for 3D scene authors to explore the many possibilities provided by VRML.

Getting started. Numerous useful resources for obtaining browser software, subscribing to the www-vrml mailing list, tutorials etc. are available via the VRML Repository (SDSC 97). Excellent books for learning VRML include (Ames 97) (Hartman 97). The VRML 97 specification is online, and an annotated version of the specification by principal VRML 97 architects Rikk Carey and Gavin Bell is available in book form and online (Carey 97). Resources on Java network programming include (Harold 97) and (Hughes 97). Intermediate and advanced textbooks for combined VRML and Java programming are (Lea 97) and (Roehl 97), respectively.

3D graphics nodes. For most programmers, there are many new language concepts and terms in VRML. An overview of this admittedly large language is necessary before describing how Java works in combination with it. This section describes the 3D-specific VRML nodes.

Big picture. Since VRML is a general 3D scene description language that can be used as an interchange file format, there are a large number of 3D graphics nodes available. These nodes are organized in a hierarchical structure called a scene graph, i.e. a directed acyclic graph. The primary interaction model for 3D VRML browsers is point and click, meaning that content can have embedded links just like the 2D HyperText Markup Language (HTML). VRML 3D browsers are typically installed as plugins within 2D browsers (such as Netscape and Internet Explorer). VRML is optimized for general 3D rendering and minimal network loading, taking advantage of extensive VRML browser capabilities. For examples, geometric primitives such as IndexedFaceSet allow authoring tools and datasets to create highly complex objects. Textures and MovieTextures can wrap 2D images over and around arbitrary geometry. Sound nodes embed spatialized audio together with the associated shapes. Lighting and camera control provide complete control of presentation and rendering, including animation of camera position to create flyby explorations or dramatic visual transitions. Finally the Script node interface to Java allows modification or generation of any VRML content.

Shape. The Shape node is a container node which collects a pair of components called geometry and appearance. Nodes that describe the objects being drawn are typically used in the geometry and appearance fields of the Shape node.

Geometry. Box, Cone, Cylinder and Sphere are nodes for simple regular polyhedra (sometimes called primitives) which provide basic building blocks for easy object construction. The Text node simplifies specification of planar or extruded polygonal text. ElevationGrid is a table of height (y) values corresponding to x-z horizontal spacing indices. The Extrusion node stretches, rotates and scales a cross-section along a spine into a wide variety of possible shapes. IndexedFaceSet, IndexedLineSet and PointSet can create 3D geometry of arbitrary complexity. Since Extrusion, IndexedFaceSet, IndexedLineSet and PointSet are specified by sets of coordinate points, color values and normal vectors can be specified point by point for these nodes.

Appearance. The appearance of geometry is primarily controlled by specifying color values or texture images. The Material node permits specification of diffuse, emissive and specular color components (which roughly correspond to reflective, glowing and shininess colors). Material nodes can also specify transparency. Colors are specified as redgreen-blue (RGB) triples ranging from 0 (black) to 1 (full intensity). Transparency value a similarly ranges from 0 (opaque) to 1 (completely transparent). As an alternative or supplement to material values, three types of texture nodes are provided. ImageTexture is the most common: a 2D image is wrapped around (or over) the corresponding geometry. MovieTexture allows use of time-dependent textures (e.g. MPEG movie files) as the image source. Finally TextureTransform specifies a 2D transformation in texture coordinates for advanced texture-mapping techniques, i.e. applying specific repetitions or orientations of a texture to corresponding geometry features.

Scene topology: grouping and child nodes. VRML syntax and node typing also helps enforce a strict hierarchical structure of parent-child relationships, so that browsers can perform efficient rendering and computational optimizations. Grouping nodes are used to describe relationships between Shapes and other child nodes. Moreover, the semantics of a scene graph carefully constrain the ways that nodes can be organized together. Child nodes come under grouping nodes to comply with the scene graph hierarchy inherent in any author's VRML scene. In addition to Shapes, child nodes describe lighting, sound, viewing, action sensors and animation interpolators. This section

synopsizes the full scope of VRML, the language; readers familiar with 3D graphics concepts may prefer skipping ahead to the Java section.

Grouping. Fundamental to any VRML scene is the notion that graphics nodes can only be grouped in ways that make sense. Grouping is used to describe spatial and logical relationships between nodes, and as an intentional side result also enable efficient rendering by 3D browsers. The Group node is the simplest of the grouping nodes: it merely collects child nodes, with no implied ordering or relationship other than equivalent status in the scene graph. The Transform node similarly groups child nodes, but first applies rotation, scaling and translation to place child nodes in the proper coordinate frame of reference. The Billboard node keeps its child nodes aligned to always face the viewing camera, either directly or about an arbitrary rotation axis. The Collision node lets an author specify a bounding box which serves as a proxy to simplify collision detection calculations for grouped child nodes. The Switch node renders only one (or none) of its child nodes, and is useful for collecting alternate renderings of an object which might be triggered by external behaviors. The LOD (level of detail) node also renders only one of multiple child nodes, but child selection is triggered automatically based either on viewer-to-object distance or on frame rate. Thus LOD enables the browser to efficiently select high-resolution or low-detail alternative renderings on-the-fly in order to support interactive rendering.

Grouping and the Web. Since the Web capabilities of VRML are analogous to HTML, two types of grouping nodes enable Web connectivity in 3D scenes. The Inline node allows importing additional 3D data from another VRML world into the current VRML world. In contrast, the Anchor node creates a link between its child nodes and an associated Uniform Resource Locator (URL) web address. When the child geometry of an Anchor node is clicked by the user's mouse, the current VRML scene is entirely replaced by the VRML scene specified in the Anchor URL. Multiple strings can be used to specify any URL, permitting browsers to preferentially load local copies before searching for remote scenes or backup locations. Typically browsers highlight "hot links" in a 3D scene by modifying the mouse cursor when it is placed over Anchor-enabled shapes.

Lighting and sound. Virtual lights in a 3D scene are used to determine illumination values when rendering. Lights cannot be "seen" in the world directly, rather they are used to calculate visibility, shininess and reflection in accordance with a carefully specified mathematical lighting model. The DirectionalLight node illuminates using parallel rays, the PointLight node models rays radiating omnidirectionally from a point, and the SpotLight node similarly provides radial rays constrained within a conical angle. Lights include color and intensity values which are multiplied against material values to calculate produce proper shading. The Sound node places an audio clip at a certain location, and with nonrendered ellipsoids surrounding it determine minimum and maximum threshold distances. Sound can repeat, be rendered spatially relative to user location, and be triggered on/off by events. Embedding various lights and sounds at different locations within a scene can produce dramatic results.

Viewing. Most 3D nodes describe location, size, shape and appearance of a model. The Viewpoint node specifies position, orientation and field of view for the virtual camera that is used to "view" (i.e. calculate) the 3D scene and render the screen image. Most objects and scenes contain a number of named viewpoints to encourage easy user navigation. The NavigationInfo node extends the camera concept to include the notion of an Avatar bounding box, used to determine camera-to-object collision and height of eye when "gravity" (i.e. terrain following) is enabled. NavigationInfo also toggles a "headlight" in the field of view for default illumination, and can switch the browser among a variety of user-navigation metaphors (FLY, EXAMINE, WALK etc.). The Fog node specifies color and intensity of obscuring fog, which is calculated relative to distance from viewer. The Background node allows specifying sky and ground color profiles, ranging smoothly from zenith to horizon to nadir. Background also permits specifying six images for texture box, which always sits beyond other rendered objects. These various viewing nodes provide rich functionality for animating viewpoint, assisting user navigation, and providing environmental effects.

Action sensors. Sensors detect change in the scene due to passage of time (TimeSensor), user intervention or other activity such as viewer proximity (VisibilitySensor). Sensors produce time-stamped events whose values can be routed as inputs to other nodes in the scene. User intervention is often as simple as direct mouse interaction with a shape via a TouchSensor, or interaction with a constraining bounding geometry specified by PlaneSensor, ProximitySensor or SphereSensor. Consistently typed input and output events are connected to correspondingly typed fields in the scene graph via ROUTEs.

Animation interpolators. Key-frame animation typically consists of simple time-varying values applied to the fields of the appropriate node. Smooth in-between animation can be interpolated linearly as long as key values are at sufficient resolution. Linear interpolators are provided for Color, Coordinate, Normal, Orientation, Position and scalar fields. Linear interpolation is sufficient for many demanding applications, including most humanoid animation. As

with Sensors, Interpolator inputs and outputs are connected with other nodes via ROUTEs. For performance reasons, use of these standard VRML sensors and interpolators is usually preferable to writing a custom script, since they can efficiently perform most authors' intended tasks. Scripts are the mechanism where authors can extend the action and animation capabilities of VRML.

Prototypes. Prototypes (PROTO and EXTERNPROTO) allow creation of new VRML node types by authoring combinations of nodes and fields from other preexisting node types. In this sense, a PROTO definition is somewhat analogous to a macro definition. In order to avoid completely copying a PROTO for each file where it is used, the EXTERNPROTO definition specifies remote URL where the original PROTO is defined, along with the interface to permit local type-checking by browsers during scene loading. The EXTERNPROTO mechanism thus allows construction of PROTO libraries for easy access and reuse.

Graphics example. No doubt dedicated readers are fully convinced by this point that VRML contains a great deal of functionality! To regain clarity, a canonical "Hello world" example in Figures 2a and 2b displays basic VRML syntax. This scene is available at www.stl.nps.navy.mil/~brutzman/vrml/examples/course/hello_world.wrl

```
#VRML V2.0 utf8
Group {
  children [
   Viewpoint {
      description "initial view"
                  6 -1 0
      position
      orientation 0 1 0 1.57
   Shape {
      geometry Sphere
                         { radius 1 }
      appearance Appearance
        texture ImageTexture {
         url "earth-topo.png"
      } }
   Transform {
      translation 0 -2 1.25
                    1 0
      rotation
                  0
                            1.57
      children [
         Shape {
            geometry Text {
          string [" Hello" "world!"]
            appearance Appearance {
              material Material {
               diffuseColor 0.1 0.5 1
         } } }
      ]
```

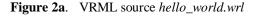




Figure 2b. VRML scene *hello_world.wrl*

VRML and **Java:** scripts, events, naming and **ROUTEs**. Interfaces between VRML and Java are effected through Script nodes, an event engine, DEF/USE naming conventions, and ROUTEs connecting various nodes and fields in the VRML scene. VRML provides the 3D scene graph, Script nodes encapsulate Java functionality, and ROUTEs provide the wiring that connects computation to rendering.

Scripts. Script nodes appear in the VRML file, encapsulating the Java code and providing naming conventions for interconnecting Java variables with field values in the scene. (Similar scripting conventions are specified for JavaScript). Interfaced Java classes import the vrml.* class libraries in order to provide type

conversion (for both nodes and simple data types) between Java and VRML. Java classes used by Script nodes must extend the vrml.node.Script class in order to interface properly with the VRML browser. The basic interface and a good description of Script nodes is excerpted from the (VRML 97) specification in Figure 3.

```
Script {
                                                           Script node is used to program behavior
  exposedField MFString url
                                              []
                                                           in a scene. Script nodes typically
  field
                            directOutput FALSE
                 SFBool
                                                           a. signify a change or user action;
  field
                  SFBool
                            mustEvaluate FALSE
                                                           b. receive events from other nodes;
  # And any number of:
                                                           c. contain a program module that
  eventIn
                  eventType eventName
                                                              performs some computation;
                  fieldType fieldName initialValue
  field
                                                           d. effect change somewhere else in the
  eventOut
                  eventType eventName
                                                              scene by sending events.
```

Figure 3. Script node specification (VRML 97).

Events. Events allow VRML scenes to be dynamic. Events are merely time-stamped values passed to and from different parts of a VRML world. EventIns accept events, and EventOuts send events (when triggered by some predefined behavior). Events must strictly match the simple type (such as integer, float, color) or node type (such as a Material node) being passed from input to output. Script parameters are designated as eventIn, eventOut or exposedField, which respectively correspond to in, out or in/out parameter semantics. Private fields are simply designated as field rather than exposedField.

DEF/USE naming conventions. Node naming and light-weight multiple instancing of nodes is possible through the DEF (define) and USE mechanisms. DEF is used to associate names with nodes. USE permits duplicate instances of nodes to be efficiently referenced without complete reinstantiation, significantly boosting performance. Node names created via DEF are also used for routing events to and from fields. Thus Script nodes (and other 3D nodes which interact with the script) all must be named using DEF. The scope of all DEF'ed names is kept local to the file (or PROTO) where the name is defined.

ROUTEs. ROUTE statements define connections between named nodes and fields, allowing events to pass from source to target. ROUTE statements usually appear at the end of a file since all nodes must be DEF'ed (named) prior to referencing. Typically ROUTEs are used for all events passed into (or out of) Script nodes. Use of ROUTEs is not always required, however, since nodes in the VRML scene can be passed by reference as fields to the encapsulated Script code. This second approach permits direct manipulation of VRML by Java without using events or ROUTEs.

Example: event-based control. A scene demonstrating VRML-Java connectivity using Scripts, events, node naming and ROUTEs is examined in Figure 4. This VRML scene and corresponding Java source code are available at www.stl.nps.navy.mil/~brutzman/vrml/examples/course/ScriptNodeEventOutControl.wrl and www.stl.nps.navy.mil/~brutzman/vrml/examples/course/ScriptNodeEventOutControl.wrl and ScriptNodeEventOutControl.ipava

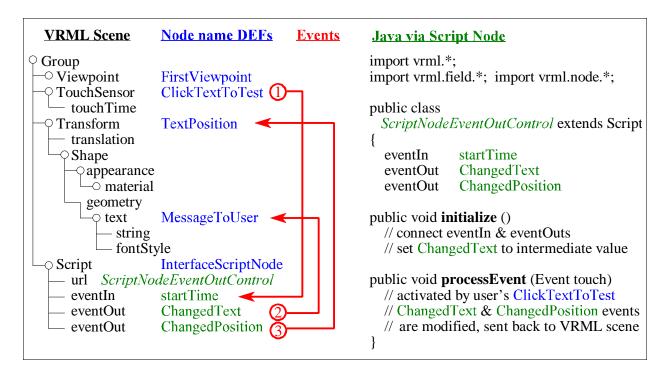


Figure 4. Script node interface between VRML and Java. This example tests event-based VRML-Java functionality. Note shared events startTime, ChangedText and ChangedPosition. The following sequence of events occurs:

- (0) Initialize method on Java side establishes links, sets trace text in 3D scene to intermediate value.
- (1) User clicks trace text in 3D scene with mouse, activating the TouchSensor built-in eventOut touchTime, which is ROUTEd to trigger the Script node EventIn startTime, which in turn invokes the processEvent() method in the corresponding Java class. Changed values for text and position are calculated by the Java class and then returned to the Script node as eventOut values.
- (2) ChangedText eventOut sent to MessageToUser text node, sets trace text in 3D scene to final message value.
- (3) ChangedPosition eventOut sent to TextPosition Transform node, moving trace text to bottom of scene.

Example: field-based control. An alternative to event passing via ROUTEs is to pass references to VRML nodes as values for fields in the Script node. In effect, Java gains direct control of VRML nodes and fields, rather than sending or receiving event messages to set/get values. Upon initialization, the Java class instantiates the node reference as a local variable. During subsequent invocations the Java class can read or modify any referenced field in the scene graph directly, without using ROUTEs. A second example follows which demonstrates the exact same functionality as the preceding example, but uses field-based control instead of events and ROUTEs. Figure 5 shows how nodes in the VRML scene are first defined, then passed as parameters to the Java class. The field-based example is available via www.stl.nps.navy.mil/~brutzman/vrml/examples/course/ScriptNodeFieldControl.wrl and ScriptNodeFieldControl.java

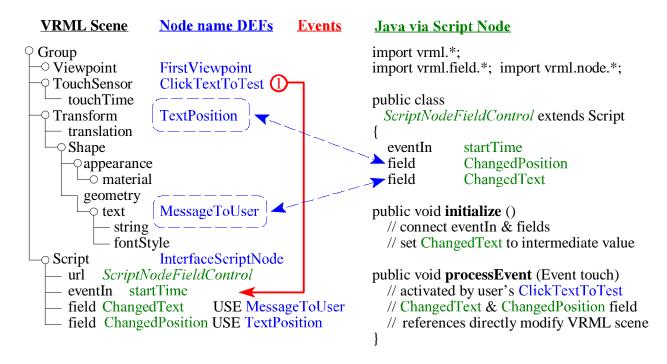


Figure 5. Field interface between VRML and Java. This example tests field-based VRML-Java functionality. Note shared event startTime, and shared fields ChangedText and ChangedPosition. Operation of this example is similar to Figure 4, except that the Java class directly manipulates VRML nodes via fields instead of sending events.

Script interface performance hints. Two authoring hints are provided as fields in the Script node for potential browser optimization: mustEvaluate and directOutput. In the first example (event-based control), mustEvaluate is set to FALSE as an author hint allowing the browser to postpone event passing until convenient, in order to optimize rendering. Similarly, the author hint directOutput is set to FALSE since the script only passes events and doesn't modify VRML nodes directly. In the second example (field-based control), the opposite values are used. Since values in the scene graph might be modified by the script directly (i.e. without notifying the browser via ROUTE activity), the hint field mustEvaluate is set to TRUE and the browser can't delay event passing as a performance optimization. Similarly, directOutput is set to TRUE to indicate that the script can modify VRML nodes directly via field control. If a scene uses both event and field control, the safest approach is to keep both values set to TRUE to maximize browser responsiveness to script actions.

Browser interface. Java via the Script node is provided a variety of methods to interact with the host Web browser. getName and getVersion provide browser identification information. getWorldURL provides a string containing the original URL for the top-level VRML scene. setDescription resets the top-level page heading. getCurrentSpeed and getCurrentFrameRate show user navigation and window redraw speeds. An author's Java program can also create and insert VRML source code (including nodes, PROTOs and additional ROUTEs) at run time. Java modifies VRML in the scene using the replaceWorld, createVrmlFromString, createVrmlFromURL, addRoute, deleteRoute, and loadURL methods. Valuable examples demonstrating these techniques appear throughout the public-domain JVerge class libraries, which provide a complete Java API mirroring all VRML nodes. JVerge accomplishes scene graph changes by sending modifications through the browser interface (Couch 97) (Roehl 97).

Future language interfaces. Java via VRML's Script node is well specified and multiple compliant browsers exist. Other interfaces are also on the horizon which can further extend Java-VRML functionality. Details follow.

External Authoring Interface (EAI). Rather than provide Java connectivity from "inside" the VRML scene via the Script node, the EAI defines a Java or Javascript interface for external applets which communicate from an "external" HTML web browser (Marrin 97). EAI applets can pass messages to and from VRML scenes embedded in an HTML page. Much of the browser interface is similar but somewhat different semantics and syntax are necessary

for event passing and flow of control. The primary benefit of the EAI is the ability for direct communications between the encapsulating HTML browser and the embedded VRML browser. The EAI will likely be proposed as an official extension to VRML 97. A next-generation Java-VRML working group is examining additional capabilities and possibly unifying the EAI with the Script node classes, for consideration in future versions of VRML.

Java3D. Sun has recently released the Java3D class library for 3D graphics programming (Deering 97). Java3D is an API, providing a programming interface for 3D that is analogous to the Abstract Window Toolkit (AWT) for 2D graphics. Java3D programs are saved as Java byte codes, not as a modeling format. Although Java3D is expected to include a loader capable of importing and exporting VRML geometry, it is not yet clear whether the Java3D event model will be able to similarly import and export VRML events. Public availability of the Java3D classes adds a number of new tools for Java programmers interested in VRML scene authoring. A VRML-Java3D working group is exploring usage conventions for interoperability, and also prototyping possible specification changes. Further unification of these already-complementary approaches holds tremendous promise.

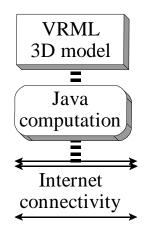


Figure 6. 3D, computation and network access provide a general entity model.

Example Java-VRML research. Programming Java in combination with VRML provides great expressive power. Simply illustrated in Figure 6, this combination of capabilities appears sufficient to provide a general entity model. VRML provides the 3D rendering and dynamic interaction capabilities, while Java provides general computation capabilities and network access. A variety of example projects follow.

Multi-user server worlds. A number of research laboratories and commercial companies are producing networked games and shared worlds that utilize centralized servers to share data among multiple participants. This alternative approach can reliably scale to several hundred participants. An extensive example that builds such a world appears in (Roehl 97).

Networking. In order to scale to many simultaneous users, peer-to-peer interactions are necessary in addition to client-server query-response. As one example, the IEEE 1278.1 Distributed Interactive Simulation (DIS) protocol is a well-specified way to pass entity behavior such as position, orientation, collision, fire/detonate and other message types. Use of multicast networking enables scalable many-to-many communications, avoiding server bottlenecks. DIS is particularly effective at communicating physics information at interactive rates in real time. The dis-java-vrml working group is implementing a public-domain DIS library in Java for use in multiple-

entity VRML worlds, available via www.stl.nps.navy.mil/dis-java-vrml

Physics of motion. Displaying realistic entity motion requires properly modeling the underlying physics which govern entity equations of motion. Much work has been done to in the graphics and robotics communities to produce kinematic (velocities only) and dynamic (forces, accelerations) models for many different types of entities. Such models typically range from three to six spatial degrees of freedom (x, y, z, roll, pitch and yaw). The NPS Phoenix autonomous underwater vehicle (AUV) hydrodynamics model provides a perhaps-worst-case example how computationally demanding physical responses can be calculated in real time (10 Hz or better on a Pentium processor). NPS AUV software is available at www.stl.nps.navy.mil/~auv. Eventually we expect that interface conventions will emerge and physics libraries for most entity types will be widely available. A particularly appealing feature of such an approach is that computational load is distributed evenly, with each entity responsible for its own physical response calculations. Conventions for kinematic human body animation are already under development by the h-anim working group at ece.uwaterloo.ca/~h-anim/.

Sound. Computational requirements for spatial audio rendering are also demanding. VRML provides a simple sound node which localizes sound clips with their corresponding geometry. Widespread work in streaming audio is beginning to provide relatively low-bandwidth protocols with adequate sound quality and scalability. Further work in spatialized audio is beginning to show that advanced techniques for aural rendering are becoming possible on desktop machines. Further information is available via the Sound in Interactive Environments (SIE) mailing list at www.cs.nps.navy.mil/people/phd/storms.

vrtp. Finally, as the demanding bandwidth and latency requirements of virtual environments begin to be exercised by VRML and Java, some client-server design assumptions of the HyperText Transfer Protocol (http) may no longer be valid. Users won't be satisfied with network mechanisms that break down after a few hundred players. A spectrum of functionality is needed on each desktop which includes client, server, peer-to-peer and network

monitoring. Our research group is building a Virtual Reality Transfer Protocol (vrtp) to better take advantage of available transport-layer functionality for VRML and overcome bottlenecks in http. Experimentation and quantitative evaluation are essential to develop the next-generation code needed for diverse inter-entity virtual environment communications.

Next steps. A great deal of implementation work is now in progress. The best news: VRML and Java are powerful software languages for 3D modeling, general computation and network access. They are well matched, well specified, openly available and portable to most platforms on the Internet. VRML scenes in combination with Java can serve as the building blocks of cyberspace. Building large-scale internetworked worlds now appears possible. Using VRML and Java, practical experience and continued success will move the field of virtual reality past speculative fiction and isolated islands of research onto desktops anywhere, creating the next-generation Web.

References

Ames, Andrea L., Nadeau, David R. and Moreland, John L., *VRML 2.0 Sourcebook*, second edition, John Wiley & Sons, New York, 1997. Information available via *https://www.wiley.com/compbooks/vrml2sbk/cover/cover.htm*

Carey, Rikk and Bell, Gavin, *Annotated VRML 2.0 Reference Manual*, Addison-Wesley, Reading Massachusetts, 1997. Available via *www.best.com/~rikk/Book/book.shtml*

Couch, Justin, *VermelGen*, software distribution, Virtual Light Company, December 1997. Available via www.vlc.com.au/JVerge

Deering, Michael and Sowizral, Henry, *Java3D Specification*, Version 1.0, Sun Microsystems Corporation, Palo Alto, California, August 1 1997. Available via *java.sun.com/products/java-media/3D/*

Harold, Elliotte Rusty, *Java Network Programming*, O'Reilly and Associates, Sebastopol California, 1997. Available via www.ora.com/catalog/javanetwk with software at ftp://ftp.ora.com/published/oreilly/java/java.netprog

Hartman, Jed and Wernecke, Josie, VRML 2.0 Handbook, Addison-Wesley, Reading Massachusetts, 1996.

Hughes, Merlin, Conrad, Shoffner, Michael and Winslow, Maria, *Java Network Programming*, Manning Publications, Greenwich England, 1997. Available via *www.browsebooks.com/Hughes* with software available on CD-ROM (due to cryptographic software export restrictions).

Lea, Rodger, Matsuda, Kouichi and Miyashita, Ken, *Java for 3D and VRML Worlds*, New Riders Publishing, Indianapolis Indiana, 1996.

Marrin, Chris, External Authoring Interface (EAI) Proposal, Silicon Graphics Inc., Mountain View California, 1997. Available via www.vrml.org

Roehl, Bernie, Couch, Justin, Reed-Ballreich, Cindy, Rohaly, Tim and Brown, Geoff, *Late Night VRML 2.0 with Java*, Ziff-Davis Press, MacMillan Publishing, Emeryville California, 1997. Information available via *ece.uwaterloo.ca/~broehl/vrml/lnvj*

San Diego Supercomputing Center (SDSC), VRML Repository, 1997, available via www.sdsc.edu/vrml

VRML Consortium (VRMLC), working groups and other information, 1997, available via www.vrml.org

VRML 97, International Specification ISO/IEC IS 14772-1, December 1997, available via www.vrml.org

About the author. Don Brutzman (<u>brutzman@nps.navy.mil</u>) is an assistant professor at the Naval Postgraduate School in Monterey California. He serves as technology vice president for the VRML Consortium.

MBone Provides Audio and Video Across the Internet

Michael R. Macedonia and Donald P. Brutzman Naval Postgraduate School

Researchers have produced the Multicast Backbone, which provides audio and video connectivity from outer space to under water -- and virtually everyplace in between. Anyone can use it.

The joy of science is in the discovery. In March 1993, our group at the Naval Postgraduate School heard that the Jason Project, an underwater exploration and educational program supported by Woods Hole Oceanographic Institution in Massachusetts, was showing live video over the Internet from an underwater robot in waters off Baja, Mexico. We worked furiously to figure out how to receive that video signal, laboring diligently to gather the right equipment, contact the appropriate network managers, and obtain hardware permissions from local bureaucrats. After several days of effort, we learned that a satellite antenna uplink cable on the Jason support ship had become flooded with seawater a few hours before we became operational.

Despite this disappointment, we remained enthusiastic because, during our efforts, we discovered how to use the Internet's most unique network, MBone. Short for Multicast Backbone [1], MBone is a virtual network that has been in existence since early 1992. It was named by Steve Casner [1] of the University of Southern California Information Sciences Institute and originated from an effort to multicast audio and video from meetings of the Internet Engineering Task Force. Today, hundreds of researchers use MBone to develop protocols and applications for group communication. Multicast provides one-to-many and many-to-many network delivery services for applications such as videoconferencing and audio where several hosts need to communicate simultaneously.

This article describes the network concepts underlying MBone, the importance of bandwidth considerations, various application tools, MBone events, interesting MBone uses (see the two sidebars), and provides guidance on how to connect your Internet site to the MBone.

Multicast networks

Multicasting has existed for several years on local area networks such as Ethernet and Fiber Distributed Data Interface. However, with Internet Protocol multicast addressing at the network layer, group communication can be established across the Internet. IP multicast addressing [2] is an Internet standard (Request For Comment 1112) developed by Steve Deering [3] of the Xerox Palo Alto Research Center and is supported by numerous workstation vendors, including Sun, Silicon Graphics, Digital Equipment Corporation, and Hewlett-Packard. Categorized officially as an IP Class D address, an IP multicast address is mapped to the underlying hardware multicast services of a LAN.

MBone and Distance Learning at the Naval Postgraduate School

Mike McCann, NPS Visualization Laboratory

In March 1993, the W.R. Church Computer Center at the Naval Postgraduate School dedicated a Sun Sparcstation 2 to act as a Multicast Backbone (MBone) router for the campus and the Monterey Bay research community. This router and an IP-encapsulated tunnel from Stanford University provides the NPS backbone with real-time audio, video, and other MBone data feeds.

The MBone is an excellent tool for those doing research in networks and video teleconferencing technology. Although it is not generally thought of as "ready for prime time" (audio dropouts may be frequent and video, at best, is only 3 frames per second over the Internet), NPS successfully used it to provide training in Cray Fortran optimization from the National Center for Atmospheric Research in Boulder, Colorado.

Five people who would not have been able to afford to travel to Boulder remotely "attended" the three-day training course at the NPS Computer Center's Visualization Laboratory. For the session, students -- including myself -- enjoyed two-way audio and video between the classroom at NCAR and the lab at NPS, and could ask questions of the NCAR instructor over the network. Advance preparation, good audio, and a camera operator in the NCAR classroom gave us a real feeling of presence in Boulder. "It was just like being there," one of my classmates said.

Paul Hyder of NCAR was instrumental in helping set up a direct "backup" tunnel between NPS and NCAR, where the slowest link is the T1 line between NPS and Stanford. During the course, there was only one 30-minute period of broken-up audio. We later determined that this interruption was caused by congestion on NCAR's Ethernet LAN. For much of 1993, the NPS Visualization Lab loaned a Sun Sparcstation 10 to the Monterey Bay Aquarium Research Institute for testing and incorporation into the live audio/video link to the research vessel *Point Lobos* and the remotely operated vehicle *Ventana* that explore the Monterey submarine canyon each day (see Figure 2). Local researchers in oceanography, virtual reality, and autonomous underwater vehicles continue to take advantage of the collaboration opportunities that this technology makes possible.

It might not be too long before MBone enables us to videoconference with a classroom or a colleague half way around the world -- directly from our desktop workstations.

Two things make multicasting feasible on a worldwide scale:

- (1) installation of high bandwidth Internet backbone connections, and
- (2) widespread availability of workstations with adequate processing power and built-in audio capability.

The reason MBone became a virtual network is that it shares the same physical media as the Internet. It uses a network of routers (mrouters) that can support multicast. These mrouters are either upgraded commercial routers, or dedicated workstations running with modified kernels in parallel with standard routers.

MBone is augmented by "tunneling," a scheme to forward multicast packets among the islands of MBone subnets through Internet IP routers that (typically) do not support IP multicast. This is done by encapsulating the multicast packets inside regular IP packets. As installed commercial hardware is upgraded to support multicast traffic, this mixed system of specially dedicated mrouters and tunnels will no longer be necessary. We expect that most commercial routers will support multicast in the near future, eliminating the inefficiencies and management headaches of duplicate routers and tunnels.

Bandwidth constraints

The key to understanding the constraints of MBone is thinking about bandwidth. The reason a multicast stream is bandwidth-efficient is that one packet can touch all workstations on a network. Thus, a 128-kilobit per second video stream (typically 1-4 frames per second) uses the same bandwidth whether it is received by one workstation or 20. That is good. However, that same multicast packet is ordinarily prevented from crossing network boundaries such as routers. The reasons for this current restriction are religious and obvious from a networking standpoint. If a multicast stream that can touch every workstation could jump from network to network without controls, the entire Internet would quickly become saturated by such streams. That would be disastrous! Therefore, controls are necessary.

MBone can control multicast packet distribution across the Internet in two ways:

- (1) It can limit the lifetime of multicast packets, and
- (2) It can use sophisticated pruning algorithms to adaptively restrict multicast transmission. This is being tested.

Responsible daily use of the MBone network consists merely of making sure you don't overload your local or regional bandwidth capacity. MBone protocol developers are experimenting with automatically pruning and grafting subtrees, but for the most part MBone uses thresholds to truncate broadcasts to the leaf routers. The truncation is based on the setting for the time-to-live (ttl) field in a packet that is decremented each time the packet passes though an mrouter. A ttl value of 16 would limit multicast to a campus, as opposed to values of 127 or 255, which might send a multicast stream to every subnet on the MBone (currently about 13 countries). A ttl field is sometimes decremented by large values under a global thresholding scheme provided to limit multicasts to sites and regions if desired.

These issues can have a major impact on network performance. For example, a default video stream consumes about 128 Kbps of bandwidth, or nearly 10 percent of a T1 line (a common site-to-site link on the Internet). Several simultaneous high-bandwidth sessions might easily saturate network links and routers. This problem is compounded by the fact that general-purpose workstation routers that MBone typically uses are normally not as fast or robust as the dedicated hardware routers used in most of the Internet.

Networking details

When a host on an MBone-equipped subnet establishes or joins a common shared session, it announces that event via the Internet Group Management Protocol. The mrouter on the subnet forwards that announcement to the other mrouters in the network. Groups are disbanded when everyone leaves, freeing up the IP multicast address for reuse. The routers occasionally poll hosts on the subnets to determine if any are still group members. If there is no reply by a host, the router stops advertising that host's group membership to the other multicast routers. MBone routing protocols are still immature and their ongoing design is a central part of this network experiment. Most MBone routers use the Distance Vector Multicast Routing Protocol, which some network researchers commonly consider inadequate for rapidly changing network topologies because routing information propagates too slowly [4]. The Open Shortest Path Working Group [5] has proposed a Multicast extension to the Open Shortest Path link-state protocol that addresses this

issue using an algorithm developed by Deering [5]. With both protocols, mrouters must dynamically compute a source tree for each participant in a multicast group.

MBone is small enough that this technique is not a problem. However, some researchers speculate that, for a larger network with frequently changing group memberships, these routing techniques will be computationally inefficient. Research efforts on these issues are ongoing, since every bottleneck conquered results in a new bottleneck revealed.

Topology and event scheduling

The MBone community must manage the MBone topology and the scheduling of multicast sessions to minimize congestion. By the beginning of 1994, some 750 subnets were already connected worldwide. Topology changes for new nodes are added by consensus: A new site announces itself to the MBone mail list, and the nearest potential providers decide who can establish the most logical connection path to minimize regional Internet loading.

Scheduling MBone events is handled similarly. Special programs are announced in advance on an MBone event electronic mail list (for example, *rem-conf@es.net* for messages and *rem-conf-request@es.net* for subscription requests) (see Tables 1 and 2). Advance announcements usually prevent overloaded scheduling of Internet-wide events and alert potential participants.

Cooperation is key. Many people are surprised to learn that no single person or entity is "in charge" of either local topology changes or event scheduling.

Protocols

The magic of MBone is that teleconferencing can be done in the hostile world of the Internet where variable packet delivery delays and limited bandwidth play havoc with applications that require some real-time guarantees. Limited experiments demonstrated the feasibility of audio over the ARPAnet as early as 1973. However, only a few years ago, transmitting video across the Internet was considered impossible. Development of effective multicast protocols disproved that widespread opinion. In this respect, MBone is like the proverbial talking dog: It's not so much what the dog has to say that is amazing, it's more that the dog can talk at all!

The key network concepts that make MBone possible are IP multicast and real-time stream delivery via adaptive receivers. For example, in addition to the multicast protocols, many MBone applications are using the draft Real-Time Protocol on top of the User Datagram Protocol and Internet Protocol. RTP [6], being developed by the Audio-Video Transport Working Group of the Internet Engineering Task Force, provides timing and sequencing services, permitting the application to adapt and smooth out network-induced latencies and errors.

Related real-time delivery schemes are also being evaluated. The end result is that even with a time-critical application such as an audio tool, participants normally perceive conversations as if they are in real time. This is because there is actually a small buffering delay to synchronize and resequence the arriving voice packets. Protocol development continues. Although operation is usually acceptable in practice, many aspects of MBone are still considered experimental.

Data compression

Other aspects of this research include the related needs to compress a variety of media and optionally provide privacy through encryption. Several techniques to reduce bandwidth include Joint Photographic Experts Group compression, wavelet-based encoding, and the ISO standard H.261 for video. Visually, this translates to "velocity compression;" rapidly changing screen blocks are updated much more frequently than slowly changing blocks.

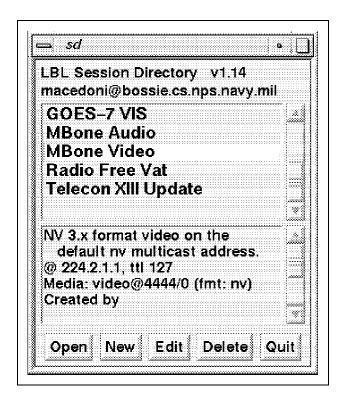
Encodings for audio include Pulse Coded Modulation and Group Speciale Mobile (the name of the standardization group for the European digital cellular telephony standard). Besides concerns for real-time delivery, audio is a difficult media for both MBone and teleconferencing in general. This is because of the need to balance signal levels for all parties, who may have different audio processing hardware (for example, different microphones and amplifiers). Audio also generates lots of relatively small packets, which are the bane of network routers.

Application tools

Besides basic networking technology, MBone researchers are developing new applications that typify many of the goals associated with the information superhighway. Session availability is dynamically announced using a tool called *sd* (session directory), which displays active multicast groups (see Figure 1). The *sd* tool also launches multicast applications and automatically selects unused addresses for any new groups. Steve McCanne and Van Jacobson of the University of California Lawrence Berkeley Laboratory developed *sd*.

Video, audio, and a shared drawing whiteboard are the principal MBone applications, provided by software packages called *nv* (net video), *vat* (visual audio tool), and *wb* (whiteboard). The principal authors of these tools are Ron Frederick of Xerox Palo Alto Research Center for *nv*, and McCanne and Jacobson for *vat* and *wb*. Each program is available in executable form without charge from various anonymous File Transfer Protocol sites on the Internet. Working versions are available for Sun, Silicon Graphics, DEC, and Hewlett-Packard architectures, with ports in progress for Macintosh. No DOS, OS-2, Amiga, or Windows versions are available, although ported tools can be found for 386 boxes running the (free) 386BSD Unix. Pointers to all public application tools are included in the Frequently Asked Questions section [1]. Mirror FTP sites are available overseas.

Additional tools are also available or under development. Winston Dang of the University of Hawaii has created *imm* (Image Multicaster Client), a low-bandwidth image server. It typically provides live images of Earth from various geostationary satellites at half-hour intervals in either visible or infrared spectra. Henning Schulzrinne of AT&T/Bell Laboratories developed *nevot*, a network voice terminal providing multiple party conferences with a choice of transport protocols. Eve Schooler of the Information Sciences Institute is part of a team developing *mmcc*, a session orchestration tool and multimedia conference control program. Mike Macedonia of the Naval Postgraduate School, coauthor of this article, has created a multicast version of NPSNET [7], a 3D distributed virtual environment that uses the IEEE Distributed Interactive Simulation (DIS)



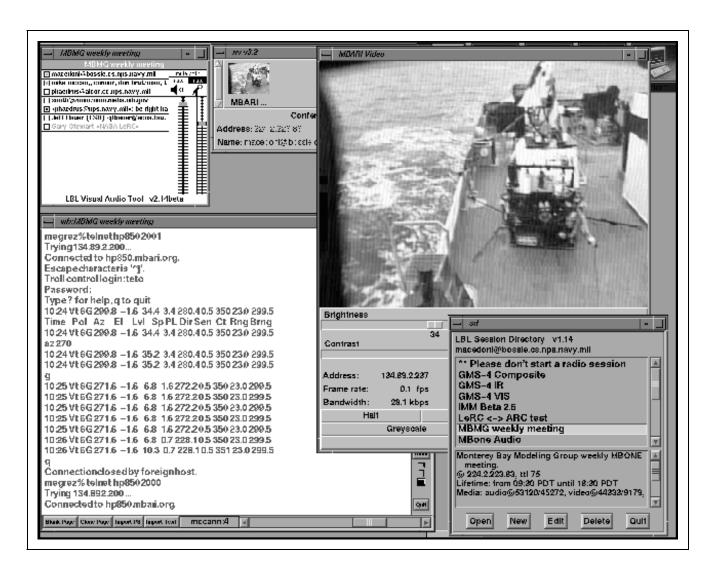
sd: session directory of MBONE events

application protocol [8]. Stephen Lau of SRI International is experimenting with using graphics workstation windows as image drivers. Kurt Lidl of UUnet Technologies, Falls Church, Virginia, is working on a network news distribution application that uses multicast to reduce overall Internet loading and expedite news delivery. (Their goal is 120 ms total propagation coast to coast -- which is amazing since light takes about 16 ms to make that trip.)

Events

Many of the most exciting events on the Internet appear on MBone first. Perhaps the most popular is NASA Select, the in-house cable channel broadcast during space shuttle missions. It's exciting seeing an astronaut positioning another astronaut by the boots to repair a satellite -- live on your desktop from 150 miles above the surface of the planet.

Conferences on supercomputing, the Internet Engineering Task Force, scientific visualization, and many other topics have appeared -- often accompanied by directions on how to download PostScript copies of presented papers and slides from anonymous FTP sites. Radio Free VAT is a community radio station whose DJs sign up for air time via an automated server (vat-radio-request@elxr.jpl.nasa.gov). Xerox PARC occasionally broadcasts lectures by distinguished speakers. Internet Talk Radio (Carl Malamud, info@radio.com) has presented talks by US Vice President Al Gore, talk-show host Larry King, and others. Another new area is remote learning, which can use MBone to bring expertise over long distances and multiply training benefits. Finally, default MBone audio and video channels are provided so that new users can experiment and get advice from more experienced users.



MBONE session with MBARI showing application tools *nv* network video, *vat* visual audio tool, *wb* whiteboard and *sd* session directory

Groupwork on groupware

The MBone community is active and open. Work on tools, protocols, standards, applications, and events is very much a cooperative and international effort. Feedback and suggestions are often relayed to the entire MBone electronic mail list. (As an example, the article you are reading was previewed by that group.)

Cooperation is essential due to the limited bandwidth of many networks -- in particular, transoceanic links. So far, no hierarchical scheme has been necessary for resolving potentially contentious issues such as topology changes or event scheduling. Interestingly, distributed problem solving and decision making has worked on a human level just as successfully as on the network protocol level. We hope this decentralized approach will continue to be successful, even with the rapid addition of new users.

Remote science over the MBone during the Jason Project

Andy Maffei, Woods Hole Oceanographic Institution

Jason/Medea is a remotely operated, dual-vehicle system developed by the Woods Hole Oceanographic Institution for underwater science and exploration. The Jason Foundation for Education uses this system as part of an annual Jason Project expedition.

During 1993, more than 600,000 K-12 students were involved in the project via live satellite transmissions. While prime broadcast hours of the expedition focused on the project's educational mission, an intense science program was conducted on a concurrent round-the-clock basis. The EDS Corporation and the University of Texas at Dallas provided a 56 Kbps data circuit for the project. Running on a Sun Workstation, MorningStar PPP software established the Internet connection with research vessel *Laney Chouest* from which the vehicles were deployed. This Internet connection made a transparent link with the multicast IP-based MBone. Although the lab experimented with multicast videoconferencing tools such as *nv* and *vat*, our primary interest in using the MBone was to transmit experimental data and to support shore-based models that depicted the positions and movement of the *Laney Chouest* and the two *Jason/Medea* vehicles. This technology was used by several investigators collaborating on Jason science projects at different locations throughout the US.

Both 2D and 3D models were developed at the Deep Submergence Laboratory for use on Sun and Silicon Graphics workstations. Software packages to access these models, along with real-time MBone data, were available to anyone on the MBone who wanted to try them out. As sonar surveys progressed during the expedition, data was transmitted back to shore, and the detailed models were updated and then distributed over the Internet.

A workstation on board the *Laney Chouest* generated multicast packets containing navigation and attitude information for the three vehicles. These packets were distributed in real time over the MBone so users running the modeling software could watch a graphic display of *Jason* prowling real seafloor features as scientists investigated seamounts, thermal plumes, and the area's unique ecology. In addition to vehicle information, experimental data variables (such as temperatures) were multicast on the MBone. Scientists and other interested users could write programs to read these experimental values, watch the models evolve, and get immediate feedback on progress being made during different experiments.

From the accounts of participating researchers, MBone use enhanced the science carried out during the cruise. However, since we spent a lot of time supporting specific experiments, we were unable to spend much time helping other interested MBone users get models up and running at their own sites. This was the first time we used multicast IP during an experiment, but we nevertheless learned a great deal. Such unique experiments demonstrate the value of other science-based tools, in addition to more generic videoconferencing applications.

For additional information, contact Andy Maffei at Woods Hole Oceanographic Institution, Deep Submergence Laboratory, Woods Hole, MA 02543.

Cost of admission

The cost of equipment is often relatively low, but to get on MBone, you need the willingness to study and learn how to use these new and fast-moving tools, you need bandwidth, and you need some hardware. NPS runs MBone tools on workstations connected via Ethernet (10 Mbps). Off-campus links are via T1 lines (1.5 Mbps).

We found that bandwidth capacities lower than T1 are generally unsuitable for MBone video, although some users -- sometimes entire countries! -- on specially configured networks have managed to make the tools work at 56 and 64 Kbps.

Given adequate network bandwidth, you next need a designated MBone network administrator. Working part-time, it typically takes one to three weeks for a network-knowledgeable person to establish MBone at a new site. Setup is not for the faint of heart, but all the tools are documented, and help is available from the MBone list.

You should read the Frequently Asked Questions (FAQ) a few times, ensure that software tools and multicast-compatible kernels are available for your target workstations, and subscribe to the mail list in advance to enable you to ask questions and receive answers. Table 1 shows the various worldwide MBone list subscription request addresses. After subscribing, review the FAQ.

Electronic mail addresses for requesting addition to the MBone mail lists.		
Electronic Mail Address For Subscription Requests	Region	
mbone-eu-request@sics.se mbone-jp-request@wide.ad.jp mbone-korea-request@mani.kaist.ac.kr mbone-na-request@isi.edu mbone-nz-request@waikato.ac.nz mbone-oz-request@internode.com.au mbone-sg-request@lincoln.technet.sg mbone-request@isi.edu	Europe Japan Korea North America New Zealand Australia Singapore Others	
	MBone mail lists. Electronic Mail Address For Subscription Requests mbone-eu-request@sics.se mbone-jp-request@wide.ad.jp mbone-korea-request@mani.kaist.ac.kr mbone-na-request@isi.edu mbone-nz-request@waikato.ac.nz mbone-oz-request@internode.com.au mbone-sg-request@lincoln.technet.sg	

Table 2.	Electronic mail addresses for putting messages on the mailing lists.	
Mail List	Electronic Mail Address for Posting Messages	Purpose
MBone	mbone@isi.edu	Network configuration and tool development
rem-conf	rem-conf@es.net	Conference announcements and general discussion

These tools can also work in isolation between workstations on a single LAN without an mrouter. We recommend that you test the application tools locally in advance (before going through the dedicated mrouter effort) to see if they are compatible with your system and match your expectations.

To receive multicast packets on your LAN, you will need to configure an mrouter. This can be either a single workstation on a LAN, or a host dedicated as a parallel mrouter. A nondedicated single workstation can receive and pass multicast to its LAN neighbors, but this arrangement can place double MBone traffic on that LAN.

A more practical approach is to dedicate an old unused workstation as an mrouter and equip it with two Ethernet cards, which are needed so this mrouter can act independently and in parallel with your standard IP router. (NPS uses both approaches.) After deciding on your mrouter configuration, obtain and load the application software tools. You are now ready to put multicast on your LAN.

Once connected, you should pass along any lessons learned to the tool authors or the MBone list. When the opportunity presents itself, show your overall network site administrator something spectacular on MBone (such as a live space walk) and make sure your site is budgeting funds to increase your network bandwidth.

Demands on network bandwidth are significant and getting more critical. You might consider Tengdin's First (and Only) Law of Telecommunications: "The jump from zero to whatever baud rate is the most important jump you can make. After that, everyone always wants to go straight to the speed of light."

Caveats aplenty

Problems still exist and a lot of work is in progress. The audio interface takes coaching and practice. Leaving your microphone on by mistake can disrupt a session since typically only one person can be understood at a time. You will need a video capture board in your workstation to transmit video, but no special hardware is needed to receive video. One-to-four frames per second video seems pretty slow (standard video is 30 frames per second), but in practice it is surprisingly effective when combined with phone-quality voice. There is one big danger: One user blasting a high-bandwidth video signal (greater than 128 Kbps) can cause severe and widespread network problems. Controls on access to tools are rudimentary and security is minimal; for example, a local user might figure out how to listen through your workstation mike (unless you unplug it). Audio broadcast preparations are often overlooked but can be just as involved as video broadcast preparations. Network monitoring tools are not yet convenient to use. There is no guaranteed delivery: Lost packets stay lost. Internet bandwidth is still inadequate for MBone in many countries.

On one occasion, an unusual topology change at NPS caused a feedback loop that overrode the NASA Select audio track. Although plenty of people were willing to point out the symptoms of our error, it was not possible for the rest of the network to cut off the offending workstation cleanly. More situations will undoubtedly occur as MBone developers and users learn more. Unpleasant surprises usually trigger a flurry of discussion and a corresponding improvement in the tools.

Expect to spend some time if you want to be an MBone user. It is time-consuming because learning and fixing are involved and because it is lots of fun!

It is not every day that someone says to you, "Here is a multimedia television station that you can use to broadcast from your desktop to the world." These are powerful concepts and powerful tools that extend our ability to communicate and collaborate tremendously. They have already changed the way people work and interact on the net.

References

- 1. Casner, "Frequently Asked Questions (FAQ) on the Multicast Backbone," May 6, 1993, ftp://venera.isi.edu/mbone/faq.txt
- 2. D.E. Comer, Internetworking with TCP/IP, Vol. 1, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
- 3. S. Deering, "Host Extensions for IP Multicasting," Request For Comment 1112, Aug. 1989, ftp://nic.ddn.mil/rfc/rfc1112.txt
- 4. R. Perlman, Interconnections: Bridges and Routers, Addison-Wesley, New York, 1993, p. 258.
- 5. J. Moy, "Multicast Extensions to OSPF," Internet Engineering Task Force Draft, July 1993, ftp://nic.ddn.mil/internet-drafts/draft-ietf-mospf-multicast-04.txt
- 6. H. Schulzrinne and S. Casner, "RTP: A Transport Protocol for Real-Time Applications," Internet Engineering Task Force Draft, Oct. 20, 1993, ftp://nic.ddn.mil/internet-drafts/draft-ietf-avt-rtp-04.ps
- 7. D.R. Pratt, M.J. Zyda, et al., NPSNET Annual Lab Review, Dec. 1993, available at ftp://taurus.cs.nps.navy.mil/pub/npsnet/npsnet.annual.report.ps
- 8. IEEE Standard for Information Technology -- Protocols for Distributed Interactive Simulation Applications, Version 2.0, Univ. of Central Florida, Inst. for Simulation and Training, Orlando, Florida, May 28, 1993.

Further reading

Baker, S., "Multicasting for Sound and Video," Unix Review, Feb. 1994, pp. 23-29.

Casner, S., and S. Deering, "First IETF Internet Audiocast," ACM SIGComm Computer Communications Review, July 1992, pp. 92-97; also as ftp://venera.isi.edu/pub/ietf-audiocast.article.ps

Curtis, P., MBone map, available via anonymous FTP from ftp://parcftp.xerox.com/pub/net-research/mbone-map-big.ps

Deering, S., "MBone: The Multicast Backbone," CERFnet Seminar, Mar. 3, 1993, ftp://parcftp.xerox.com/pub/net-research/cerfnet-seminar-slides.ps

Acknowledgments

We thank the originators of the MBone tools and dozens of MBone users who provided essential contributions to this article.

Michael R. Macedonia is a US Army major and a PhD student in computer science at the Naval Postgraduate School. His research is directed toward the development of software architectures supporting large-scale distributed virtual environments. During Operations Desert Shield and Desert Storm, he led a team that designed and established computer communications between the Joint Electronic Warfare Center modeling network and US forces in the Middle East to provide electronic warfare analytical products for combat commanders.

Macedonia received a BS from the US Military Academy in 1979 and an MS from the University of Pittsburgh in 1989. He is a member of the IEEE Computer and Communications Societies.

Donald P. Brutzman is a lieutenant commander and submarine officer in the US Navy, and has served three submarine tours. His current doctoral research investigates the design and construction of an underwater virtual world that includes an operational autonomous underwater robot. His research interests also include 3D real-time computer graphics, virtual worlds, scientific visualization, underwater robotics, distributed simulation, machine learning, and high-performance network applications. He is editor of video proceedings for annual unmanned underwater vehicle conferences.

Brutzman received a BSEE from the US Naval Academy in 1978 and an MS in computer science from the Naval Postgraduate School in 1992. He is a member of IEEE, the IEEE Computer Society, ACM, and the American Association for Artificial Intelligence.

Readers can contact the authors at the Computer Science Department, Naval Postgraduate School, Monterey CA 93943-5000. Their e-mail addresses are *macedonia@cs.nps.navy.mil* and *brutzman@nps.navy.mil*

FTP Availability

This article is available electronically via anonymous FTP to *taurus.cs.nps.navy.mil* in subdirectory *pub/mbmg* as *mbone.ps*; in Uniform Resource Locator (URL) format used above, this anonymous FTP request corresponds to

ftp://taurus.cs.nps.navy.mil/pub/mbmg/mbone.ps

Text and hypertext versions of this article are also available as ftp://taurus.cs.nps.navy.mil/pub/mbmg/mbone.txt ftp://taurus.cs.nps.navy.mil/pub/mbmg/mbone.html

This article was originally published in **IEEE COMPUTER** magazine, pp. 30-36, April 1994.

Exploiting Reality with Multicast Groups: A Network Architecture for Large-scale Virtual Environments

Michael R. Macedonia, *Michael J. Zyda, David R. Pratt, Donald P. Brutzman, Paul T. Barham
Computer Science Department
Naval Postgraduate School
Monterey, California 93943-5118 USA
+1-408-656-2305
{macedonia, zyda, pratt, brutzman, barham}@cs.nps.navy.mil

ABSTRACT

We describe a network software architecture for solving the problem of scaling very large distributed simulations. The fundamental idea is to logically partition virtual environments by associating spatial, temporal, and functionally related entity classes with network multicast groups. We exploit the actual characteristics of the real-world large-scale environments that are simulated by focusing or restricting an entity's processing and network resources to its area of interest via a local Area of Interest Manager (AOIM). Finally, we present an example of how we would implement this concept for ground vehicles. We have begun design and construction of the AOIM for use with the NPSNET 3D vehicle simulator. NPSNET is currently the only Distributed Interactive Simulation (DIS) protocol compliant simulator using IP Multicast communications and is suitable for operation over the Internet.

KEYWORDS: Virtual Reality, Distributed Interactive Simulation, Internet Protocol Multicast, Distributed Interactive Entertainment, Large-scale Virtual Environments.

NETWORKED VIRTUAL WORLDS

The development of multi-user networked virtual worlds has become a major area of interest to the graphics community. The realization of high bandwidth wide area communications, the success of World Wide Web applications such as the National Center for Supercomputing Application's Mosaic browser, and government funding of Distributed Interactive Simulation (DIS) has fueled the desire to expand networked virtual worlds beyond local area networks. However, the Internet has proved a challenging environment for real-time applications such as interactive virtual worlds and multimedia.

Our group has been motivated to expand the capabilities of simulations and virtual environments (VEs) by exploiting multicast networks to serve medium to large numbers (more than 1,000) of simultaneous users. In particular, distributed interactive entertainment applications such as multiplayer games, whether in-home or location-based, will require scalable network architectures in order to provide both rich environments and profitable returns. This paper outlines the problems of scaling the most influential 3D distributed VE architectures, DIS and SIMNET, and our work in developing a solution to the design and construction of large-scale distributed simulations. In particular this paper addresses the networking software architecture for large-scale virtual environments (VEs).

Computer graphics applications are fundamentally concerned with visual bandwidth, i.e. the transfer of visual information to a human user. Visual bandwidth factors include image complexity, rendering speed, interaction latency and computational complexity. In this paper we examine how distributed applications can best exploit global network connectivity in order to drive the visual bandwidth of interactive 3D graphics for large-scale virtual environments.

PRACTICAL PROBLEMS WITH THE DIS PROTOCOL

Advances in computer architectures and graphics, as well as standards such as the IEEE 1278 Distributed Interactive Simulation (DIS) and BBN SIMNET protocols have made small scale (today, less than 300 interactive players) realistic man-in-the-loop simulations possible [1]. These standards have been used by the military for several years. Unfortunately, SIMNET, which was developed for small unit training, and its descendant, DIS, are currently not suitable for large-scale multiplayer VEs.

The following is a list of several major problems associated with scaling the current suite of DIS protocols in order to illustrate the difficulties of building large-scale VEs:

Enormous bandwidth and computational requirements for large-scale simulation. In schemes such as SIMNET and DIS, a simulation with 100,000 players could require on the order of hundreds of megabits per second of network bandwidth to each computer participating in the simulation, an unrealistic requirement for an affordable system in this decade. Maintaining the state of all other entities, particularly with dead-reckoning algorithms (which use second-order kinematics equations), will be a major bottleneck for large-scale simulation. Recent experiences with the U.S. Army's Synthetic Theater of War (STOW) exercises have shown this to be the case [10].

Faster computers and networks will not necessarily satisfy these needs. First, faster networks require faster processors merely to copy packets from the network into user space even before the application touches the protocol data unit (PDU). Second, the creeping demand for more realism (i.e. collision detection and constraint satisfaction) will introduce a rapid rise in computational and spatial complexity with even modest size VEs [5].

We conjecture that on the order of 1000 entities are the upper bound limit to which a single host can realistically manage in real-time despite future advances in computer and graphics architectures because of computational complexity.

Multiplexing of different media at the application layer. The current DIS protocol requires the application to multiplex and demultiplex different types of real-time data (e.g. simulation packets, audio, and video) at the application layer rather than at the network or transport layers. Therefore, the virtual environment must treat continuous video streams identically to bursty simulation traffic, i.e. through allocation of buffers and timing at the application layer.

Lack of an efficient method of handling static objects. Large numbers of static entities such as bridges and buildings may change with respect to an event (e.g. an explosion). These and other stationary objects must send update messages at regular intervals to inform the participants of their current state. For example, a tank that has been destroyed must constantly inform the world that it is dead to inform new entrants or other entities that may have missed the original state change message.

Models and world databases must be replicated at each simulator. No mechanism in DIS exists to distribute objects on demand. For large-scale simulation this is a necessity, particularly when the simulators are heterogenous, controlled by different organizations, and minimal coordination is expected prior to an exercise. Furthermore, it is neither feasible nor efficient for each simulator to store every model and database for a 100,000 entity simulation. For example, a human simulation (e.g. a dismounted infantryman) on land normally does not need to concern itself with naval vessels, unless some unique scenario has the human near enough to the ocean so that it is visible.

REASONS FOR PROBLEMS

Event and State message paradigm. A basic requirement for DIS has been that the simulation of the VE must be, as a whole, stateless -- data is fully distributed among the participating hosts and entities are semi-persistent. Therefore, every entity must be made aware of every event (e.g. a missile detonation communicated by a Detonation Protocol Data Unit or DPDU) just on the chance it may need to know it. According to the protocol, an entity must, on a regular basis, communicate all of its state information (an Entity State Protocol Data Unit or ESPDU) to every member of the group - even though the data contained in the ESPDU is often redundant and unnecessary (e.g. aircraft markings). More importantly, these "keep alive" ESPDU messages can comprise 70% of the traffic for large-scale simulations.

This paradigm as applied in DIS does not take into consideration that different simulated systems have different real-world sensing capabilities that translate into each entity's VE data requirements. In a large VE, it is unlikely that two entities representing ground vehicles separated by 200 km need to be aware of each other. Yet, under the current architecture they must inform each other of state changes and updates.

The rationale for this is to avoid the reliability problems of a central server, to simplify communication protocols, and minimize latency while guaranteeing that hosts entering a simulation would eventually build their entity database through entity state and event messages. Furthermore, the use of broadcast ESPDU updates is part of the effort to maintain consistent view among the simulators within a particular tolerance.

Real-time system trade-off's. Reliability (guarantees that data sent is received) normally is compromised for real-time performance in large distributed groups. This is because in order to be truly reliable the system requires the use of acknowledgment schemes such as the one used in Transport Control Protocol (TCP) which defeats the notion of real-time, particularly if a player host must establish a virtual connection with every other entity host to ensure that each received data correctly. Therefore, large-scale environments must rely on connectionless (and therefore unreliable) network protocols such as the User Datagram Protocol (UDP) for wide-area communications.

The corollary is that a real-time environment should avoid transactions between individual entities since this requires reliable communications. Furthermore, schemes that use a central database do not work well in a large VE due to I/O contention. For example, AT&T's Imagination network limits the number of concurrent players in a game to four because they are centrally served and bandwidth is limited to the speed of modems (less than 28 Kbps).

No "middleware" layer. There does not exist a DIS protocol component that mediates between distributed VE applications and the network. The current DIS paradigm implies the use of a bridged network because every message is broadcast to every entity. However, internetworking (routing over the network layer) is necessary for large-scale simulations because it provides the capability to use commercial services as opposed to private networks to bring together diverse, geographically dispersed sites; use different local network topologies and technologies (e.g. Ethernet and FDDI); and take advantage of "rich" topologies for partitioning bandwidth, providing robustness and optimization of routes for minimizing latency. Confining DIS to the data link layer requires the use of bridges which are an order of magnitude slower to reconfigure after a topological change than routers while the number of stations are limited to the tens of thousands. A network with routers is limited to the numbers accommodated by the address space.

Origins as small unit training systems for Local Area Networks (LANs). Many of these problems devolve from the fact that until recently DIS and SIMNET were used exclusively for small scale training simulations. In this mode it has been relatively easy to insure that the VE components have homogenous sets of models and terrain databases by replicating them at each host. The lack of middleware stems from the monolithic nature of these small scale environments which could be distributed using a single LAN. Hence, *broadcast* communication was sufficient for these limited environments.

These origins have also influenced the current assumptions about the density and rates of activity of entities in large-scale simulations that do not necessarily match the real world. Players in SIMNET participated for short periods (several hours) and were highly active because the purpose of the simulation was to train crews in coordinated drills. Furthermore, the density of entities with respect to the simulated area of play was high because that best represented a small unit engaged in close combat and because of the difficulty in using large terrain data bases.

The SIMNET experience contrasts sharply with real large-scale exercises. Figure 1 depicts the location of some of the 2191 entities from an actual combat training scenario at the U.S. Army National Training Center (NTC), Ft. Irwin, CA and replayed in the Janus Combat Model. Our analysis from this exercise showed that in the ten hours of total maneuver, one third of the vehicles did not move. As the exercise progressed, over half of the vehicles became disabled and stopped all movement. Furthermore, 60% of the terrain was outside the detection range of all the vehicles.

Figure 1. An armored brigade in the attack at the National Training Center, Ft. Irwin, CA. The area is 60 x 50 km. Friendly vehicles are blue; enemy vehicles are red.

EXPLOITING REALITY

Increasing the number of entities by more than two orders of magnitude requires us to think beyond these artificial situations. We believe that it is incorrect to strictly extrapolate the SIMNET and DIS experience (or any of the small-scale research VEs) to large-scale VEs. Moreover, large VEs are likely to be domain specific in their requirements. We can exploit aspects of the real-world such as areas of interest and movement rates to efficiently use multicast groups, eliminate ESPDU keep-alive updates, enhance the reliability of large-scale VEs, and reduce overall bandwidth requirements.

In the real world, which virtual environments emulate, entities have a limited area of interest. For example, a tank on a battlefield can effect and observe other entities out to a range of less than 4 km. On the other hand, a person on foot typically has an area of interest of only several hundred meters. This would be the case for a dismounted infantryman or a human simulated for a typical role-playing adventure game. The entities whose areas of interest overlap are members of a *spatial class* or group in the VE.

With respect to the military domain, group membership within these classes would change relatively slowly. Helmbold in his study on the rates of advance rates for land operation found that land combat operations stand still 90-99% of the time [2]. The worlds record for aggregate movement in modern warfare was 92 km/day for 4 days (or about 6 km/hour) by the 24th Mechanized Infantry Division in Desert Storm [3]. Individual vehicles may move much faster, but they would not continue at high rates very long because they fight as part of units in which movement must be coordinated.

RELATED WORK

The partitioning of virtual worlds into spaces is a common metaphor for VEs. Multi-User Dungeons (MUDs) have used this idea and projects like *Jupiter* from Xerox PARC have extended this to associating "rooms" with multicast video and audio teleconferences [7]. Benford has described a concept for the spatial interaction of objects in a large-scale VE [8]. The spatial model uses different levels of awareness between objects based on their relative distance and mediated through a negotiation mechanism. An implementation using DIVE (Distributed Interactive Virtual

Environment) uses "standard VR collision detection" to determine when the transitions between awareness levels should occur [9]. The MASSIVE project also uses this approach. However, the need for collision detection, reliable communication, and strong data consistency have made it difficult for DIVE and MASSIVE to scale beyond a handful of users [8]. This may be changing as their developers pursue the use of multicast communications and weaker data consistency.

APPROACH

Our approach is computationally efficient--constant time versus $O(\log n)$ for simple collision detection using octrees or bounding volumes--and takes advantage of multicast networks for partitioning the environment [4]. Additionally, we consider two other criteria for establishing relevance among entities and their communication in the VE.

Entities also may belong to a *functional class* in which an entity may communicate with a subset of entities. Therefore, simulated radio traffic should be restricted only to the interested parties of the group. Other types of functional classes could be related to system management or services such as time synchronization.

Another example of a functional class in the military domain would be a VE "air control" group. The group would include entities that are primarily concerned with entities or events occurring in the air. Therefore, air defense and aircraft entities would comprise the majority of the group. Aircraft and air defense systems are relatively sparse in the whole as compared to other combat systems such as tanks. Air defense systems would also belong to a small subset of the spatial class. Aircraft which are interested in a particular area of ground can "focus" and join a spatial group associated with its area of interest.

Finally, entities can belong to a *temporal class*. For example, some entities do not require real-time updates of all state changes. A system management entity might only need updates every several minutes. Similarly, a simulator of a space-borne sensor only needs a general awareness of ground vehicle entities and therefore can accept low-resolution updates. When there is a need for more resolution, the simulator, like aircraft entities, can focus and become part of a spatial group.

DIS AREA OF INTEREST MANAGER

We propose the use of a software "glue" between the DIS event and state PDU paradigm and the network layers that is wedded to reality. The area of interest manager (AOIM) partitions the VE into a set of workable, small scale environments or classes to reduce computational load on hosts, minimize communications on network tail links, and localize reliability problems. Furthermore, the AOIM exists with every simulator to distribute partition processing among hosts.

MULTICAST

The AOIM uses spatial, temporal, and functional classes for establishing membership in multicast network groups as shown in Figure 2. Multicast services allow arbitrarily sized groups to communicate on a network via a single transmission by the source. Multicast provides one-to-many and many-to-many delivery services for applications such as teleconferencing and distributed simulation in which there is a need to communicate with several other hosts simultaneously. For example, a multicast teleconference allows a host to send voice and video simultaneously to a set of (but not necessarily all) locations. With broadcast, data is sent to all hosts while unicast or point-to-point routes communication only between two hosts.

The Internet Group Management Protocol (IGMP) provides an addressing scheme for an unreliable, connectionless, multicast service that is routable over the Internet [4]. From the perspective of the AOIM, IP Multicast allows the creation of transient multicast groups that can be associated with an entity's area of interest (AOI).

In this context, IP Multicast addresses can essentially be used as context labels instead of physical destinations. Figure 3 shows this. Players X, Y, and Z send data to the IP Multicast group address 224.11.22.56 rather than explicitly forwarding packets to each and every player. The network takes over this requirement. Players A and B send and receive traffic relevant only to their group, 224.11.22.33, while C is a member of both and participates in each session.

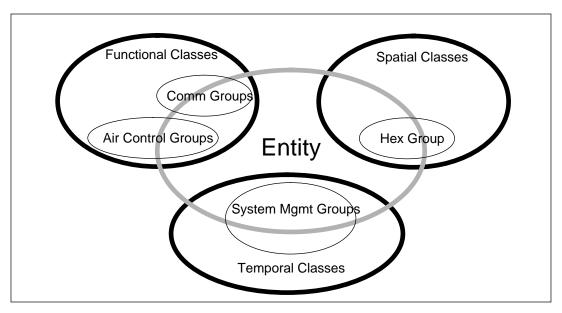


Figure 2. Relationship between entity and multicast groups.

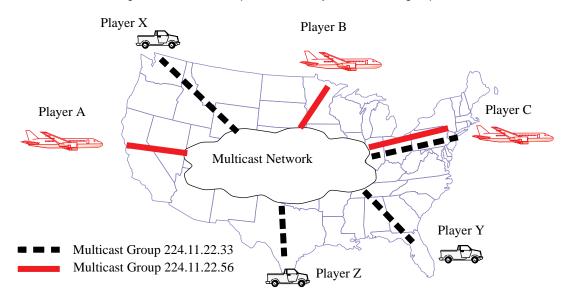


Figure 3. Simple illustration of multicast communications. Groups are expressed as IP Multicast Addresses. Note that Player C is a member of both multicast groups

Therefore, multiplexing and demultiplexing is done at the network level. This naturally provides a way of separating classes of traffic such as audio, video and simulation data. For example, the radio communications functional class would be mapped to a particular multicast group address or "channel" group (Figure 4). Filtering of appropriate multicast groups occurs in the network interface hardware and does not consume processor cycles.

As stated before, such partitioning is necessary to reduce the enormous computational requirements of large-scale (100,000 player) simulations. For a 1000 object exercise conducted in 1990 with SIMNET, the limiting factor was not network bandwidth, with loads running at 50%, but the local host processor performance.

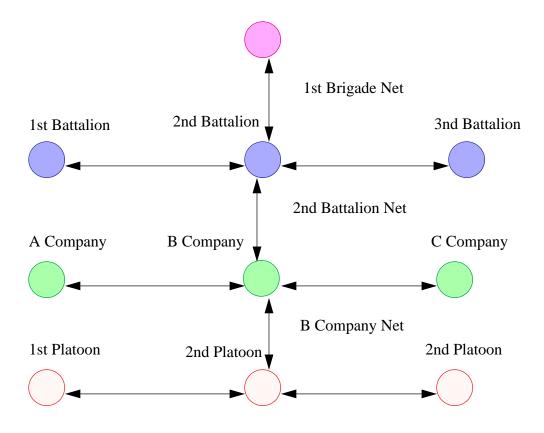


Figure 4. Military communication networks.

ASSOCIATIONS

To illustrate our ideas, we examine using the AOIM to associate spatial classes with multicast addresses. We suggest for this example partitioning the VE with appropriately sized hexagonal cells. Each cell is mapped to a multicast group. In Figure 5 we associate a vehicle with seven cells that represent its AOI. Hence, it is also a member of seven network multicast groups. The entity's host listens to all seven groups but, with two exceptions, it sends PDUs only to the one associated with the cell in which it is located.

There are several reasons hexagons are used. First, they are regular, have a uniform orientation, and have uniform adjacency. As the vehicle moves through the VE, it uniformly adds and deletes the same number of cells/multicast groups. A vehicle's AOI is typically defined by a radius - much like signal of transmitter in a cellular telephone system. If squares were used, we would either need to include more area than was necessary (and thus include more entities in our AOI) or use smaller grids - requiring more multicast groups - and compute which grids the vehicle should be associated with.

GROUP CHANGES

Entities can belong to several groups at a time to avoid boundary or temporal aliasing. There will likely be few group transitions by a ground-based entity within an hour because, on average, groups of vehicles will move slowly relative to the entire VE. If a vehicle was moving at the Desert Storm record advance rate, it would transition on average a cell once an hour. The vehicle portrayed in Figure 5 must join and leave three multicast groups which are associated with cells at the periphery of its AOI where change is less critical - ameliorating the effects of latency caused by joining and leaving new groups. The outlined clear cells are removed and the outlined grey cells are added as the entity transitions to a new cell.

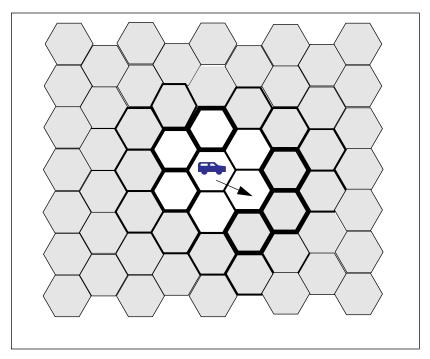


Figure 5. Area of Interest for vehicle mapped to a subset of multicast groups.

We use group changes as an opportunity for database updates -- similar to a paged memory scheme -- in order to eliminate regular ESPDU updates. We do this in a logical, distributed manner using knowledge about the age of entities with respect to their particular group.

An entity joins a group as a passive or active member. Active members send as well as receive PDUs within the group, are located in the cell associated with the group, and can become the group leader. Passive members normally do not send PDUs to the group except when they join or leave. They are associated with the group because the cell is within their AOI, yet they are not located within the cell.

When an entity joins a new group it notes the time it entered and issues a *Join Request PDU* to the cell group. The PDU has a flag indicating whether it is active or passive. The group leader replies with a *Pointer PDU* that references the request and in turn multicasts a PDU containing a pointer to itself or another active entity. The new member sends a *Data Request PDU* to the referenced source which issues a *Data PDU* containing the aggregate set of active entity PDUs. A passive entity becomes an active member of a group by reissuing the *Join Request PDU* with a flag set to active when entering a cell. Departures from the group are announced with a *Leave Request PDU*.

We use the oldest member of the group as the election method for group leader. We make use of timestamps to determine the oldest member. The first active member of a group will issue several Join Request PDUs before concluding that its is the sole member of the group and therefore the oldest. When a passive entity determines that there is no leader, it merely listens for active members. A new active member of an established group issues a Join Request PDU, receives the Data PDU, notes the join timestamps of the members, and keeps track of those who enter and leave.

RATIONALE

The Data PDU may be sent reliably to the issuer of the Join Request PDU via a unicast protocol as a heavy-weight object. With a large member distributed simulation, reliability, as provided in the Transmission Control Protocol (TCP), would normally penalize real-time performance merely by having to maintain timers for each host's acknowledgment. Moreover, flow control is also not appropriate for DIS since systems with humans in the loop can recover from a lost state message more gracefully than from late arrivals. Fortunately, within the context of DIS, a certain amount of unreliability is tolerable and is mediated through the use of the dead-reckoning and smoothing

algorithms. Other applications such as packet voice and video can use adaptive techniques to handle lost packets and delays. However, we can reliably send the Data PDU because the entity will normally be joining a group that is at the periphery of its AOI where latency is not as critical. Learning about new members of entity groups under the DIS model takes at least five seconds while transiting through the VE to a new active group. Assuming sufficient bandwidth, new entrant learning can take less than a one second under our architecture.

Furthermore, large-scale VEs will naturally have some degree of unreliability. Currently, an entire DIS simulation involving hundreds of entities can fail due to a single rogue application because all communication is broadcast. In the DIS exercises it has been possible for a malfunctioning device or application to "jam" the simulation. Partitioning the VE into groups prevents problems from impacting on the entire simulation.

The AOIM can be run as a separate thread or process and eliminates the need to change current DIS PDU semantics. The upper-level application simulating an entity is not required to have knowledge of the partitioning. Therefore, many current DIS applications can be adapted to support this architecture.

COMMUNICATIONS MODEL

We conjecture that a large-scale real-time VE cannot guarantee strong data consistency and reliable communication among all its participants simultaneously. Instead, four types of communication can be established which, used together, allow stronger consistency than simply broadcasting state messages. They provide for a much richer world through a mechanism for sending large objects reliably and supporting VE partitioning.

In our model there exists four methods for communication within the context of VEs:

Light-weight interactions. These messages are composed of the same state, event, and control PDUs used in the DIS paradigm but implemented with multicast. They are light-weight because the complete semantics of the message are encapsulated within the maximum transfer unit (MTU) of the underlying data link to permit asynchronous real-time interactive use. Therefore, these PDUs are not segmented. They are either received completely or not at all because they are communicated via connectionless and unreliable (unacknowledged data) networks. The MTU values are 1500 bytes for Ethernet and 296 bytes for 9600 Point-to-Point (PPP) links.

Network pointers. Proposed are light-weight references to resources, in a way similar to Uniform Resource Identifier (URI) as defined in the Hypertext Transfer Protocol (HTTP). Pointers are multicast to the group so that they can be cached by members. Therefore, common queries need not be resent and the server can direct responses to other members of the group. We make a distinction between pointers and light-weight interactions (e.g. Join Request PDU) because they do not completely contain an object but rather its reference. Pointers provide a powerful mechanism for referencing not only the current aggregate state of the group but also terrain, model geometry, and entity behaviors defined by a scripting language. In the context of the World-Wide Web (WWW), network pointers have revolutionized Internet communication.

Heavy-weight objects. These objects require reliable, connection-oriented communication. For example, an entity may require model geometry after joining a group that does not exist in its database. The entity would multicast a request for the geometry and the response would be a multicast pointer to the source. The originating entity then receives the heavy-weight object via a reliable network connection. If efforts such as the Virtual Reality Modeling Language (VRML) are successful, heterogeneous systems may be able to exchange this type of information.

Real-time streams. Video and audio traffic provide continuous streams of data that require real-time delivery, sequencing and synchronization. Moreover, these steady streams will be long-lasting, persisting from several seconds to days. They are multicast on a particular "channel" to a functional class. In contrast with the current DIS protocol, we propose the use of pointers which direct entities to these channels rather than forcing the VE (which may be as simple as a text-based application) to receive both light-weight DIS PDUs as well as video streams. Moreover, the VE can spawn a separate process which incorporates an adaptive receiver and thereby separates the handling of bursty simulation message from real-time streams.

STATUS OF WORK

We have developed an IP Multicast version of the NPSNET-IV 3D vehicle simulator using a network library developed by Paul Barham and John Locke that supports multiple threads and dynamic creation of multicast groups [6]. Furthermore, we are incorporating the algorithms to support the AOIM and have developed a simulation to predict and evaluate the results. NPSNET is widely used by universities, industry, and government for distributed virtual environment research (Figure 6). NRL is using NPSNET to explore large-scale DIS. We are also collaborating with Sarcos Inc. and the University of Pennsylvania to insert humans into VEs to support medical emergency and soldier skill training.



Figure 6. Medic conducting first-aid in NPSNET

The multicast version of NPSNET has been successfully tested over the Internet with several sites including Naval Research Laboratory (NRL), George Mason University, MIT, Sprint, Stanford Research Institute and the Rand Corporation. We have also modified a version of the Modular Semi-Automated Forces (MODSAF) simulator developed by Loral to support multicast for the generation of large numbers of simulated military vehicles and aircraft.

SIMULATION

The simulation developed was used to examine these questions:

- Does partitioning using our architecture reduce bandwidth and computational requirements for large-scale VEs as compared to the DIS model?
- Does the architecture scale and if so how well?

We were interested in the effect the behavior of entity distribution and maneuver had on our architecture --particularly as compared to the current DIS broadcast scheme. The simulation was developed by collecting entity behavior data from Janus, a large constructive model widely used by the US Army for research and training. The data was based on a real-world military scenario (similar to one proposed for use in STOW 97) and actual large-scale exercises at the NTC. After post-processing, we then took the military entity data and applied our partitioning algorithms.

Our simulation using spatial partitioning shows that, in a military context, as the number of entities increase with our architecture, the mean peak bandwidth was less than T-1 rates (Figure 7). The largest peaks are primarily caused by the transfer of large data objects when entities transition among groups. This will be quite feasible over networks to the home or office in the near future. For example, AT&T and Intel are planning to convert cable systems to provide 28 Mbps bandwidth to the home [11].

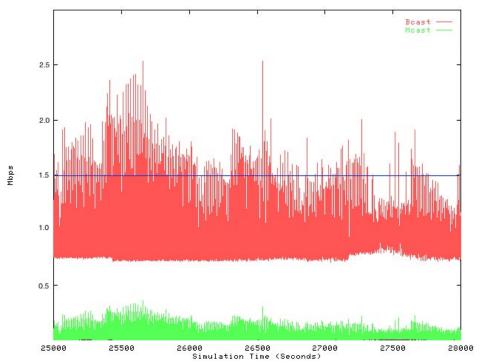


Figure 7. Mean multicast (4 km hex) vs. broadcast traffic with 2191 entities.

The AOI architecture takes advantage of the fact that it does not use entity keep-alives or heartbeats for new entrant learning to reduce the bandwidth costs associated with them. Rather, new entrants are informed of the existence of other entities during the Join procedure. Furthermore, assuming that entities are distributed across different subnets, multicast association reduces the traffic demands on tail links by confining the scope of an entity's communication to its area of interest and *implicitly* directing its traffic to a subset of hosts on the network.

Perhaps more important than ameliorating bandwidth costs, partitioning can reduce the amount of state that an entity must maintain. The architecture, as opposed to the DIS model, scales with the increase in entities. For our simulation, using the NTC data with four km radius's hexes, the *maximum* number of entities in an area of interest was relatively constant at approximately 1800 entities as we increased the number of entities from 2191 to four times that amount (Figure 8). This number probably represents the worst case with a mixture of tanks and light infantry with their weapon systems and vehicles.

The peak number of entities using the architecture presents a feasible computational goal. Moreover, we can reduce the maximum AOI density by making our hexes smaller or reduce the impact by doing application level filtering.

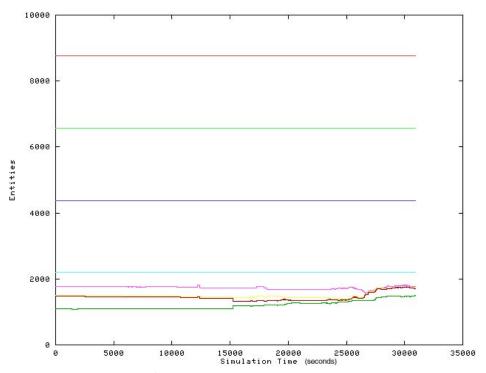


Figure 8. Comparison of max entities in AOI.

LIMITATIONS OF THE WORK

This work does not address all the problems of building large-scale VEs in a military environment. First, this architecture may complicate developing secure environments because encryption devices may need to authenticate every other device for each multicast address. Second, our work has not analyzed the impact of fast moving entities such as aircraft. We conjecture that this will not be a major obstacle for a number of reasons. Most aircraft fly too high or too fast to actually observe individual ground entities and therefore establish an association with them except for air defense systems. In the case of a system like JSTARS which tracks ground vehicles, we suggest that it would belong to the functional "air" group and could receive low rate Entity State PDUs from the temporal "all" group.

For example, all ground entities could send an ESPDU every time it had moved five km or every hour to the all group. For fifty thousand entities this is roughly thirteen PDUs per second. Low flying aircraft like helicopters must normally hover or circle to acquire a target and fly at a fifth the speed of fighters. Therefore, these aircraft can join the spatial groups associated with their target area. These actions and the effects of other types of entity behavior needs exploration.

Third, we did not directly consider network topology in our simulations. We need to determine whether this architecture may be more appropriate for a network with many subnets with a single entity or host located at the site versus one with a handful of subnets with hundreds of entities represented on each host. Our data suggests the former and in the future we need to use models such as those being developed by NRL to examine this issue. We also need to examine the impact of other partitioning methods such as functional partitioning. In particular, we have not tested the impact of multimedia communication using this architecture though we have used voice and video in conjunction with NPSNET.

CONCLUSION

This paper describes a concept that provides a network software architecture for solving the problem of scaling very large distributed simulations. The fundamental idea behind our approach is to logically partition virtual environments by associating spatial, temporal, and functionally related entity classes with network multicast groups. This is accomplished by exploiting the actual characteristics of the real-world large-scale environments that are to be simulated, and by focusing an entity's processing and network resources to its area of interest via an Area of Interest Manager.

ACKNOWLEDGMENTS

This work would not have been possible without the support of our research sponsors: USA ARL, ARPA, DMSO, USA STRICOM, USA HQDA AI Center-Pentagon, USA TRAC.

- 1. Institute of Electrical and Electronics Engineers, International Standard, ANSI/IEEE Std 1278-1993, *Standard for Information Technology, Protocols for Distributed Interactive Simulation* (March 1993).
- 2. Helmbold, R. L. *Rates of Advance in Historical Land Combat Operations*, technical report, CAA-RP-90-1, US Army Concepts Analysis Agency, Bethesda, MD, (June 1993), pp. 5-2,3.
- 3. McQuie, R. *Historical Characteristics of Combat for Wargames*, technical report, CAA-RP-87-2, US Army Concepts Analysis Agency, Bethesda, MD, (July 1988), p. 13.
- 4. Macedonia, M.R., and Brutzman, D.P., MBone Provides Audio and Video Across the Internet. *IEEE Computer*, 27, 4 (April 1994), pp. 30-34.
- 5. Pentland, A.P., Computational Complexity versus Simulated Environments. *Computer Graphics. 1990 Symposium on Interactive 3-D Graphics* 24, 2 (March 1990), 185-192.
- 6. Macedonia, M. R., Zyda M.J., Pratt D.R., Barham P.T., Zeswitz S. NPSNET: A Network Software Architecture for Large-scale Virtual Environments. *Presence* 3,4 (Winter 94).
- 7. Curtis, P., Nichols, D.A. MUDs Grow Up: Social Virtual Reality in the Real World. 1994. ftp://ftp.parc.xerox.com/pub/MOO/papers/MUDsGrowUp.ps
- 8. Benford, S., Bowers, J., Fahlen, L. and Greenhalgh, C. Managing Mutual Awareness in Collaborative Virtual Environments. In *Proceedings of VRST '94*, World Scientific Publishing Company, NJ, pp. 223-236.
- 9. Carlsson, C. and Hagsand, O. (1993). DIVE a Multi User Virtual Reality System. In *Proceedings of VRAIS* '93 (September 18-22, Seattle, WA) IEEE, NJ, 1993. pp. 394-400.
- 10. Solitaire, Robert. STOW-E To Do List. (1 December 1994). Debriefing slides.
- 11. "On-ramp for info highway is closer", San Jose Mercury News, 5 May 1995, D1.

"MBone Unplugged"

- or -

how we created a wireless mobile television studio and multicast SIGGRAPH 95 world-wide over the Internet.

WHAT, WHEN and WHERE was SIGGRAPH 95?

The annual conference of the Association for Computing Machinery (ACM) Special Interest Group on Graphics (SIGGRAPH) includes over 30,000 computer experts, explorers, and enthusiasts from all over the world. All focused on the adventurous edge of cyberspace, interactive digital techniques, new entertainment media, networked communities, and the science that creates tomorrow's technologies.

August 6-11 1995, Los Angeles Convention Center, Los Angeles California USA.

Our Objectives

- Create a mobile studio for a weeklong KSIG-TV. SIGGRAPH exhibitors were our programs.
- Provide content and lab facilities for local educators to learn about the Internet.

Here is a video clip from local television station KCBA. They interviewed Tracey Emswiler about our efforts. The 50-second clip is a 19.2 MB QuickTime video.



Enter the Cart...

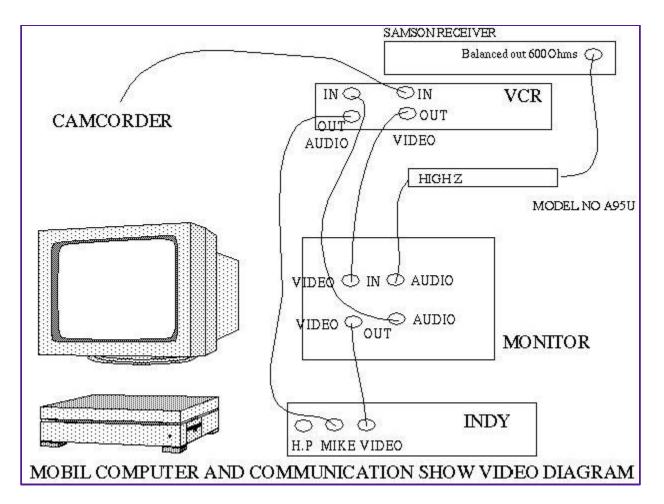
Here is the "MBone Unplugged" cart Jon Bigelow put together:



The cart was WAY COOL! It included the following gear:

- Silicon Graphics Inc. (SGI) entry-level *Indy* workstation
- AirLAN wireless bridges (2 Mbps bandwidth, spread spectrum, ~ 1 GHz frequency band, no FCC license required)
- Uninterruptible Power Supply (UPS), good for about 15 minutes
- · Standard VHS VCR and television monitor
- · Two VHS videocameras with tape capability and single mobile tripod
- Wireless microphones, lapel and hand-held
- Switch box to select camera/audio source
- Video projector to put live imagery on any nearby wall
- · LOTS of wires, cables and adapters, all labeled at each end
- · Two cellular telephones for coordination and mobility
- Broomstick to lift wireless antenna above cart
- · Various NPS master's students and passersby to push the darn thing

Here is a simplified wiring diagram.



This is the **workstation of the future**. It has everything: computing, graphics, live 2-way audio/video, mobility and global scope. We all want one now!

Undoubtedly someone can make a much more compact version. The usefulness of this approach is perhaps the most important lesson we've learned.

Network Considerations

<u>GraphicsNet</u> was SIGGRAPH 95's high-performance, state-of-the-art communications infrastructure. It included ATM, HIPPI and Ethernet networks. We are particularly grateful for the help and expertise of the GraphicsNet team.

Multicast Backbone (MBone)

Here are links to an article on the <u>Multicast Backbone (MBone)</u> and an <u>local MBone home page</u> pointing to other free software and information resources.

The following are a few of the MBone issues we grappled with. More will be added.

We announced our intention to multicast during the week of 6-11 August 95 to the remote conferencing list (rem-conf@es.net). and also to the MBone Session Agenda scheduling home page at http://www.cilea.it/MBone/agenda.html. There were no objections to our plans.

We intentionally stressed the MBone as much as possible, remaining within the community guidelines regarding permissible use. Our motivation was that the most interesting problems and fixes on the MBone often occur when someone is "pushing the envelope" of current capabilities. Originally we expected to share global MBone bandwidth with the NASA Select channel. Since the space shuttle mission was cancelled, we were able to use 256 Kbps video

bandwidth as opposed to 128 Kbps video (the ordinary default). It was very helpful not to have to share bandwidth. Some coordination on site was neccessary with the GraphicsOnline team, but few difficulties occurred.

We were glad to see that the MBone held up well. There were occasional problems with users elsewhere inadvertantly sending high-bandwidth video, but these were quickly detected and corrected, either by Van Jacobson (University of California, Lawrence Berkely Laboratory) or the offending users themselves. This is a long-standing training problem. It has been alleviated somewhat by making transmission a little harder fro for naive users to turn on. Nevertheless we think it is a people problem which needs a people solution (e.g. simple effective training).

mrouted difficulties and recovery

In addition to the cart, we brought an old Sun 3/110 workstation to serve as the multicast router. Unfortunately there is a tiny switch on the back which we inadvertantly moved from "NORMAL" to "DIAGNOSTIC." This meant it would not boot and was effectively inoperable. If we had a flashlight, we might have figured out what we were doing wrong. As it turned out, **Sun Microsystems** loaned us a Sparc 20 workstation as a replacement mrouter. Greg Earle of NASA JPL reconfigured the kernel and saved us. Dan Evans of NASA Ames provided the encapsulated multicast tunnel from Moffett Field.

On several occasions our mrouter stopped sending multicast due to bugs in the *vic* video tool crashing on the Sun mrouter. At the same time the machine appeared to be operational based on normal routing and continuous ping checks. This condition was hard to diagnose given our minimal use of multicast diagnostic tools. Better tool usability is needed. We are also considering putting the multicast router and tunnel endpoint right on the mobile machine. As long as it has enough cycles to operate, complete system state is right there for the user to evaluate. Our distance from the mrouter made it difficult to determine if our mrouter or an mrouter upstream from the convention center was the cause of MBone connectivity losses. As it turned out, all of our difficulties were tracable to our GraphicsNet mrouter.

The production environment at SIGGRAPH was too noisy and dynamic to permit monitoring e-mail. We were able to get effective out-of-band communications from remote viewers by modifying the name field in the video-audio tool (vat).

We used the net video (*nv*) software to multicast video. Our motivation for using *nv* was to permit another workstation to act as a reflector to CU-SeeMe. Unfortunately our CU-SeeMe expert never showed, and we didn't have time to learn it ourselves in mid-conference. In general, *vic* in .H261 mode is superior. H261 (part of the MPEG standards suite) is an asymmetric encoding, meaning that more computational work is performed by sender while higher framerate and lower effective bandwidth is enjoyed by the receivers. Previous tests showed that our Indy was able to handle this load effectively.

We used *vat* audio software. Standard settings were employed (encoding PCM2). Modifications included setting Lecture mode for longer playouts, "Mike mutes Net" to prevent offsite interuptions, and use of the user's Name field to list each event by name as they occurred.

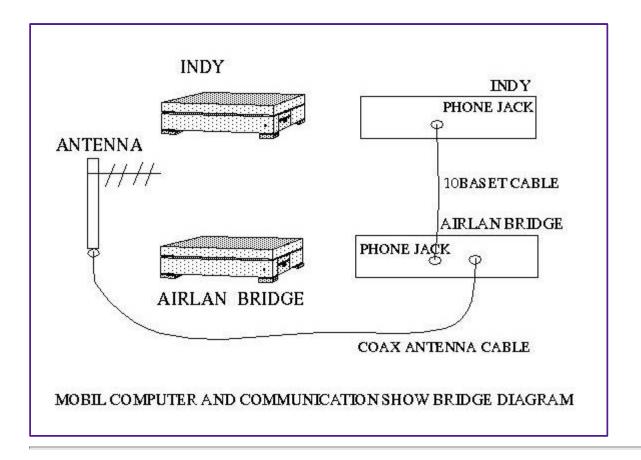
It is of course very important to make sure someone is keeping track of the videotape being recorded so that new ones are started as soon as they are needed.

Tracey Emswiler was in charge of event scheduling and coordinating the transitions to the next venue. It was a big help having a single person keep everything on track.

Wireless considerations

The AirLAN bridges are very slick. Called "smart bridges," they operate by listening for IP numbers on either side. Once the bridges know that a host is on one side or the other, any packets appearing on the opposite side are automatically sent across the bridge. Thus each side looks like a singe local-area network (LAN) without manual route reconfigurations. We found them to be reliable in a mobile environment, and the frequency agility of a spread spectrum approach made the link very robust despite lots of "trash RF" in this very crowded exhibition hall environment.

This diagram shows how the wireless bridges were connected.



Educator Training in Monterey



Teacher training is an ongoing project for us. K-12 collaboration has been particularly productive in helping us learn which technologies work for people and which don't. Click on the map above to learn more about our regional projects.

These are the home pages we created for Monterey Bay use during this event.

- Flyer: http://www.stl.nps.navy.mil/~tlemswil/flyer.html
- What's happening: http://www.stl.nps.navy.mil/~tlemswil/
- Schedule: http://www.stl.nps.navy.mil/~tlemswil/schedule.html
- Top 14 List For Researchers, Educators and Kids: http://www.stl.nps.navy.mil/~tlemswil/ResearchersTop 14.html
- Top 10 List For Water Discovery Institute Teachers: http://www.stl.nps.navy.mil/~tlemswil/WaterDiscoveryTop 10.html

More about our efforts in networked K-12 education:

 <u>Networked Ocean Science Research and Education, Monterey Bay California</u>, available at <u>http://inet.nttam.com/HMP/PAPER/039/</u>

- Learning About Monterey Bay (LAMBAY) http://lambay.cse.ucsc.edu/mb
- Monterey Bay Regional Education Futures (MBReEF) Consortium http://www.ucsc.edu/mbay-region
- Initiative for Information Infrastructure & Linkage Applications (I3LA) <u>ftp://taurus.cs.nps.navy.mil/pub/i3la/i3la.html</u>
- Real-time Environmental Information Network & Analysis System (REINAS) http://csl.cse.ucsc.edu/reinas.html

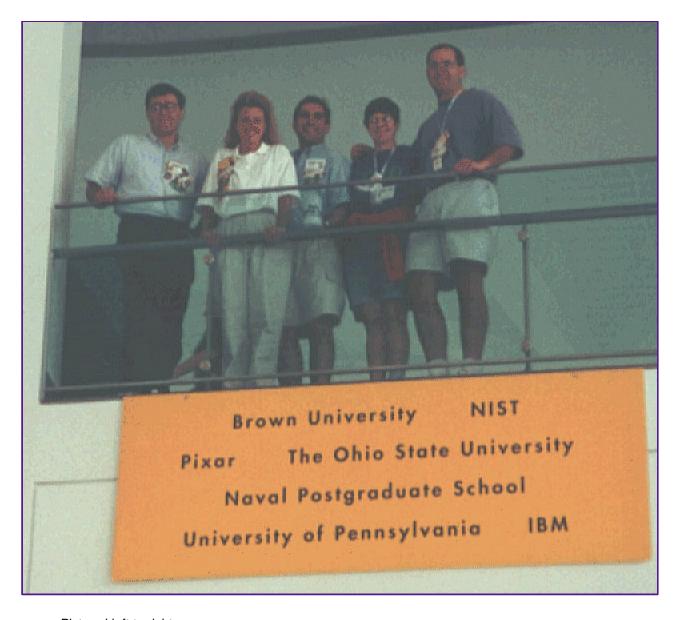
Future Work

We are still thinking over lessons learned and future work. Many of these issues will likely become theses for masters students in the NPS Information Infrastructure Research Group (IIRG). Queries are welcome.

- · digitally archive recorded videotapes for MBone-compatible access on demand
- · record a full week of K-12 Internet sessions at INET 96, Montreal Canada
- · do more cool educational stuff for free
- · learn new answers and new questions
- cost-benefit comparisons between building custom videoteleconferencing (VTC) classrooms and mobile networked VTC carts
- · do the same thing at much higher bandwidth over ATM
- show that Macs and PCs can be used for this, once they get multicast support built into the operating system kernel
- distribute the software tools and technical exertise to regional K-12 schools when the technology is ready

People

The "Unplugged" (unhinged?) Team in LA:



Pictured left to right:

- Don Brutzman (UW/Br) (brutzman@nps.navy.mil) AUV / IIRG / NPSNET research groups
- <u>Tracey Emswiler</u> (ITM) (<u>tlemswil@nps.navy.mil</u>) <u>Hamming</u> "Learning to Learn" <u>Multicast</u> (graduated September 95)
- Jon Bigelow (ITM) (rjbigelo@nps.navy.mil) Monterey BayNet (graduated August 95)
- Beth Bacon (bbacon@attmail.com) Writer and communications consultant
- Dan Bacon (CS) (<u>bacon@cs.nps.navy.mil</u>) Adding a physically based submarine to NPSNET (graduated September 95)

and also in LA (but not pictured)

- John Sanders (<u>JoSanders@wposmtp.nps.navy.mil</u>) NPS Public Affairs Officer (PAO)
- LT Clyde Wetteland USN (<u>CWetteland@mntry.nps.navy.mil</u>) NPS Assistant Public Affairs Officer (PAO)

Back in Monterey helping the teachers and students and researchers:

- Dennis Trepanier (ITM) (dmtrepan@nps.navy.mil) Regional K-12 network management (graduated September 95)
- Matthew Koebbe (Dr. Longhair) (phaedrus@nps.navy.mil) NPS Visualization Lab

- CDR Bob Ellis (bellis@cs.nps.navy.mil) Computer Science Curriculum Officer
- Charles Peyton Taylor (ctaylor@nps.navy.mil) Root 262 Macintosh Lab Manager
- Katie Muir (kmuir@mbayag.org) Monterey Bay Aquarium (MBA) Water Discovery Institute
- Ray McClain (mcclain@mlml.calstate.edu) Moss Landing Marine Laboratory Researchers' Invitational
- Jeff Forte (CS/ITM)(<u>jeforte@nps.navy.mil</u>) ATM, dual degree CS/ITM (September 96)
- <u>Michael Tiddy</u> (ITM) (<u>metiddy@nps.navy.mil</u>) (September 96)

Thanks

- ACM SIGGRAPH
- · Coco Conn and Zane Vella, Interactive Communities and Digital Circus
- GraphicsNet team including chair Marke Clinger (Fore Systems), Peter Haddad (Hewlett-Packard), Jeannette Dravk and Craig Schell and Amy Wong (Fore Systems) and the rest of the gang
- · Garry Paxinos (MetroLink Inc.) and the SIGGRAPH OnLine team
- Virtual Reality Modeling Language (VRML) group
- Diane Sena, Steve Webster, Jeff Bryant and Katie Muir (Monterey Bay Aquarium MBA)
- Peter Brewer and Bruce Gritton (Monterey Bay Aquarium Research Institute MBARI)
- John Morales, Frank Cardoza, Harry Thomas (NPS Electronic Media Division) and Tracy Hammond (NPS Registrar)
- Mike Zyda, Dave Pratt, Mike Macedonia and the NPSNET group
- Dan Evans and Greg Earle (NASA Ames and JPL)
- Van Jacobson and the MBone list participants
- Theresa-Marie Rhyne (Lockheed Martin) and the "Future Technologies" panel
- Ray McClain (MLML) and Bob Ellis (NPS)
- Terry Williams, Milena Cochran, Stefan Hudson and Gary Porter (NPS Systems Technology Lab)
- Mike McCann and Matthew Koebbe (NPS Visualization Lab)
- Mike Newman (Newman and Associates)
- · Silicon Graphics Inc. (SGI) and Sun Microsystems
- I3LA Monterey Bay Educators
- · Hotel Nikko, Beverly Hills
- Murat Tamer for the wiring diagrams
- and everyone else who helped that we've inadvertantly forgotten.

T-Shirts and Buttons

These are the most competitive measure of SIGGRAPH value. We got t-shirts, buttons and a few paper sacks stuffed with unmarked bills from the following folks:

- SIGGRAPH 95, SIGGRAPH 96 and ACM
- GraphicsNet
- Silicon Graphics Inc. and the OpenInventor group
- Sun Microsystems
- · Journal of Graphics Tools
- Mark Pesce, author of the new book "VRML: Entering CyberSpace" from New Riders Press
- Hard Rock Cafe
- Ed Debevic's Good Eats (including my favorite, "I'd Rather Be Bowling")

Who OWES us shirts (we have long memories):

Parallax Graphics

Still to work on:

MBone monitoring tools, lessons learned from Monterey side, ATM demo with Parallax, individual sessions we

recorded to tape digitized and placed online, etc.

Authors: Don Brutzman and Tracey Emswiler

"MBone Unplugged" home page: <u>Don Brutzman (brutzman @nps.navy.mil)</u> (31 January 96)